Intelligent Sound Processing

S_TOOLS-STx USER’S GUIDE

ACOUSTICS RESEARCH INSTITUTE
AUSTRIAN ACADEMY OF SCIENCES

DRAFT: NOT FOR PUBLICATION

16 February 2002
Manual produced by:
Peter Balazs, Werner A. Deutsch, Anton Noll and Jonathan White

Edited by: John R. Rennison

© (2001)
Acoustics Research Institute
of the Austrian Academy of Sciences, Vienna.
What is STx? – Intelligent Sound Processing

Contents

1. WELCOME TO STX – INTRODUCTION ................................................................. 1
  1.1 WHAT IS STX? – INTELLIGENT SOUND PROCESSING ............................ 1
      1.1.1 Signal Processing ................................................................. 2
      1.1.2 Batch Mode and Dynamic Data Exchange (DDE) .................... 2
      1.1.3 Macro Programming ......................................................... 3
      1.1.4 Compatibility .................................................................. 3
      1.1.5 Sound Presentation System ............................................. 3
      1.1.6 Local and Distributed Sound Archive Applications .......... 3
      1.1.7 LAN Implementation .......................................................... 4
      1.1.8 Internet Service ................................................................. 4
  1.2 INSTALLING STX FOR WINDOWS .............................................................. 5
      1.2.1 System Requirements .......................................................... 5
      1.2.2 PC Audio System Requirements ..................................... 5
      1.2.3 Program Installation .......................................................... 5
      1.2.4 Online Help and How to Contact the STx Team ................ 6
  1.3 PRINCIPLES OF DIGITISATION AND SOUNDFILES .................................. 6
  1.4 SOUNDFILE AND SOUNDSEGMENT MANAGEMENT: BASIC CONCEPTS ... 7
      1.4.1 Soundfiles .............................................................. 8
      1.4.2 Soundsegments (Audio Segments) ..................................... 8
      1.4.3 Soundsegment Directory Files ......................................... 9
      1.4.4 Soundsegment List ........................................................ 9
      1.4.5 Soundsegment (Audio Segment) Definition ..................... 9
  2. STX OVERVIEW ................................................................................................. 10
     2.1 THE STX WORK AREAS .................................................................... 10
          2.1.1 Startup ............................................................................ 10
          2.1.2 The STx Main Application Window .................................. 11
          2.1.3 The S_TOOLS Application – the Soundfile Work Area .... 12
     2.2 RECORD – START/STOP RECORDING .............................................. 13
          2.2.1 Play/Record Setup ............................................................ 15
          2.2.2 Input Signal Level Control (Metering) ............................ 16
          2.2.3 Dynamic Range ............................................................. 17
     2.3 CREATING TAGS AND CUE POINTS DURING RECORDING .................. 17
     2.4 PLAY – REPRODUCE ......................................................................... 18
  3. SOUNDFILES AND SOUNDSEGMENTS IN STX ........................................... 21
     3.1 FINDFILE – ADD SOUNDFILES TO THE WORK AREA ....................... 21
          3.1.1 Folders/Files Summary .................................................. 23
          3.1.2 Advanced Findfile Options ............................................ 23
     3.2 THE SOUNDFILE LIST WINDOW ......................................................... 24
Welcome to STx – Introduction

3.2.1 Remove a Soundfile from the Work Area ...................................................... 25
3.2.2 Automatic Saving of the Work Area ............................................................ 25
3.2.3 Delete a Soundfile from Disk ................................................................. 25
3.2.4 Save File List (As) - Load File List ......................................................... 25
3.3 SOUND SEGMENT LABELLING AND METADATA MANAGEMENT .................. 26
3.3.1 What Are Soundsegments? ........................................................................ 26
3.3.2 How to Specify and Manipulate Soundsegments ....................................... 27
3.3.3 Segments – New: Creating an Entry for a Segment (Name) ......................... 28
3.3.4 Segments – Delete: Erase a Segment ....................................................... 29
3.3.5 Segments – Sequence: Open the STx Sequencer ...................................... 29
3.4 TAKING CARE OF YOUR SOUNDSEGMENTS ........................................... 29
3.5 SOUNDSEGMENT DATABASE (IN PREPARATION) ...................................... 30

4. STX WAVEFORM EDIT AND TIME-DOMAIN VIEW ......................................... 31
4.1 PRELIMINARIES ............................................................................................. 31
4.2 WAVEFORM DISPLAY AND ANALYSIS (VIEWER1) ................................. 31
4.2.1 Built-in Display Types .............................................................................. 31
4.2.2 Custom Displays and Analyses ............................................................. 33
4.2.3 Signal Addressing ................................................................................... 33
4.3 WAVEFORM VIEWER1 LAYOUT ................................................................. 33
4.3.1 Cursor Functions and Hotkeys .............................................................. 34
4.3.2 Signal Playback ..................................................................................... 34
4.3.3 Cursor Navigation .................................................................................. 36
4.3.4 Waveform Zoom Windows .................................................................... 38
4.3.5 Segment Naming, Tagging and Annotation ............................................ 39
4.3.6 Viewer Cue Point Lists (currently inactive) ............................................. 40
4.4 CONFIGURATION OF WAVEFORM VIEWERS .......................................... 40

5. STX SPECTRUM ANALYSIS .............................................................................. 43
5.1 STX SPECTRUM VIEWER2 ........................................................................ 43
5.1.1 Pre-configured Viewer2 Settings ........................................................... 43
5.1.2 Analyse Startup Sequence ..................................................................... 45
5.1.3 Analysis Parameter Settings .................................................................. 46
5.2 FFT NARROWBAND SPECTROGRAMS (MUSIC) ........................................ 48
5.3 FFT WIDEBAND SPECTROGRAMS (SPEECH) .......................................... 49
5.4 CEPSTRAL ANALYSIS .................................................................................. 50
5.5 LINEAR PREDICTION ANALYSIS ............................................................. 52
5.6 FORMANT FREQUENCY CONTOUR IN SPEECH ..................................... 54
5.7 PSEUDO WIGNER DISTRIBUTION .......................................................... 54
5.8 FREQUENCY BAND ANALYSIS .................................................................. 56
5.9 SHORT-TIME SPECTRUM AND SPECTRUM AVERAGING ....................... 57

6. SIGNAL PROCESSING OF STX SOUND-SEGMENTS ....................................... 59
6.1 SIGNAL COPY ............................................................................................... 59
What is STx? – Intelligent Sound Processing

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.2</td>
<td>AMPLIFY / LIMIT / NORMALIZE</td>
<td>61</td>
</tr>
<tr>
<td>6.3</td>
<td>AUTOMATIC GAIN CONTROL (AGC)</td>
<td>61</td>
</tr>
<tr>
<td>6.4</td>
<td>DIGITAL FILTERING: LP-HP-BP, SPECTRUM</td>
<td>62</td>
</tr>
<tr>
<td>6.4.1</td>
<td>Low-pass Filter</td>
<td>62</td>
</tr>
<tr>
<td>6.4.2</td>
<td>High-pass Filter</td>
<td>62</td>
</tr>
<tr>
<td>6.4.3</td>
<td>Bandpass Filter</td>
<td>62</td>
</tr>
<tr>
<td>6.4.4</td>
<td>Spectrum-(Multi-band-) Filter</td>
<td>62</td>
</tr>
<tr>
<td>6.5</td>
<td>DENOISING (SPECTRAL SUBTRACTION)</td>
<td>63</td>
</tr>
<tr>
<td>6.6</td>
<td>IRRELEVANCE FILTER “WHAT YOU SEE IS WHAT YOU HEAR!”</td>
<td>64</td>
</tr>
<tr>
<td>6.7</td>
<td>OVERMASKING (SPECTRAL SUPPRESSION)</td>
<td>64</td>
</tr>
<tr>
<td>7.1</td>
<td>Creating a Segment Sequence</td>
<td>65</td>
</tr>
<tr>
<td>7.2</td>
<td>Managing and Using Segment Sequences</td>
<td>67</td>
</tr>
<tr>
<td>7.3</td>
<td>Creating a Sequence of Synthesized Sounds</td>
<td>67</td>
</tr>
<tr>
<td>8.1</td>
<td>DATA SET HANDLING AND EXPORT OF PARAMETERS</td>
<td>69</td>
</tr>
<tr>
<td>8.2</td>
<td>WORKING WITH DATA SETS</td>
<td>70</td>
</tr>
<tr>
<td>8.2.1</td>
<td>Speech Parameters are stored in the Data Set</td>
<td>71</td>
</tr>
<tr>
<td>8.2.2</td>
<td>Export of Parameter Values after Editing (Data Set)</td>
<td>73</td>
</tr>
<tr>
<td>9.1</td>
<td>DEFINING SOUNDSEGMENTS</td>
<td>75</td>
</tr>
<tr>
<td>9.2</td>
<td>SPECIFYING SOUNDSEGMENTS ON THE FLY</td>
<td>76</td>
</tr>
<tr>
<td>9.2.1</td>
<td>Soundsegment Addressing throughout All STx Modules</td>
<td>76</td>
</tr>
<tr>
<td>9.3</td>
<td>SOUNDSEGMENT DIRECTORY FILE TEMPLATE(S)</td>
<td>77</td>
</tr>
<tr>
<td>9.4</td>
<td>FREQUENCY RESOLUTION &amp; WINDOW LENGTHS</td>
<td>78</td>
</tr>
<tr>
<td>9.5</td>
<td>LONG-DURATION SEGMENT ANALYSIS</td>
<td>79</td>
</tr>
<tr>
<td>9.6</td>
<td>WAVEFORM DISPLAY – VIEWER 1</td>
<td>79</td>
</tr>
<tr>
<td>9.7</td>
<td>ANALYSIS – VIEWER 2</td>
<td>80</td>
</tr>
<tr>
<td>9.8</td>
<td>AUDIO LEVELS AND AMPLITUDE MEASUREMENTS</td>
<td>81</td>
</tr>
<tr>
<td>9.8.1</td>
<td>Voltage and Audio Signal Levels</td>
<td>81</td>
</tr>
<tr>
<td>9.8.2</td>
<td>Amplitude Measurements, Root Mean Square Value, VRMS</td>
<td>82</td>
</tr>
<tr>
<td>9.9</td>
<td>PREFERRED FREQUENCIES AND FREQUENCY BANDS</td>
<td>84</td>
</tr>
<tr>
<td>9.10</td>
<td>FREQUENCIES OF MUSICAL NOTES</td>
<td>85</td>
</tr>
<tr>
<td>9.11</td>
<td>MUSICAL INTERVALS IN CENTS</td>
<td>86</td>
</tr>
<tr>
<td>9.12</td>
<td>HOTKEYS</td>
<td>87</td>
</tr>
<tr>
<td>9.12.1</td>
<td>Waveform Viewer1 Hotkeys</td>
<td>87</td>
</tr>
<tr>
<td>9.13</td>
<td>RANDOM NUMBER GENERATORS</td>
<td>88</td>
</tr>
<tr>
<td>9.13.1</td>
<td>MathCad</td>
<td>88</td>
</tr>
<tr>
<td>9.14</td>
<td>GLOSSARY</td>
<td>89</td>
</tr>
<tr>
<td>9.15</td>
<td>SOME EXAMPLES OF SPEECH ANALYSIS</td>
<td>93</td>
</tr>
</tbody>
</table>
10. TABLE OF FIGURES ............................................................ 97
1. Welcome to STx – Introduction

1.1 What is STx? – Intelligent Sound Processing

STx is the Windows version of the well-established S_TOOLS workstation for acoustics, speech and signal processing, which was first developed under DOS using the AT&T DSP32 signal processor card. STx now runs on 600 MHz PC systems equipped with Accelerated Graphics Port (AGP) cards and standard sound cards. It performs nearly all the functions of the former DSP hardware configuration and is under continuous development. Today STx provides tools for:

- Digitisation of sound (analog to digital transfer)
- Segmentation, storage, archiving and migration of large soundfiles
- Visualisation, labelling and annotation of sound data in the time domain
- Frequency analysis
- Visualisation, labelling and annotation of time-varying signals by means of spectrograms (in the frequency domain)
- Wigner distribution and other time-frequency representations
- Cepstrum analysis
- Linear Prediction Coding (LPC) analysis
- Extraction of formant frequency candidates (in speech)
- Fundamental Frequency Analysis
- Adaptive filtering (digital filter techniques)
- Denoising, signal enhancement
- Auditory perception modelling (Computational Hearing)
- Irrelevance Filtering (adaptive filtering and matching to the masking of human hearing: Relevance Spectrograms)
- Modified Phase Vocoder implementation
- Tracking of perceptually relevant spectral components by overmasking (figure – background discrimination)
Welcome to STx – Introduction

- Tools for time segmentation
- Sound object segmentation and sequencing
- Phonetics and Music analysis tools
- Metadata generation for speech corpora, music and sound collections

By introducing a Windows version of S_TOOLS, we have achieved: a) independence of hardware configurations and b) a remarkable reduction (of more than 50%) in the cost of hardware. At the moment, the real-time graphics are still subject to some minor limitations, but improvement can be expected from future developments in PC and graphics subsystem technology.

1.1.1 Signal Processing

STx incorporates a sophisticated sound analysis and signal processing package:

- frequency analysis based on Fast Fourier Transformations (FFT)
- further spectrum estimation methods, e.g. AR, Pseudo-Wigner distribution
- fundamental frequency and formant frequency extraction (cepstrum, LPC)
- digital filter implementations
- hearing test software (some already provided, more in preparation)

Working with STx is highly effective because

- STx is specifically designed to handle thousands of large binary sound objects in the range of hundreds of GBytes
- a real-time 2-channel frequency analyser is included
- the creation of publication-quality graphs is supported.
- STx can communicate with external programs such as word processors, graphics, mathematics and statistical packages on an ASCII file basis

1.1.2 Batch Mode and Dynamic Data Exchange (DDE)

STx supports batch mode on most I/O and analysis functions. For a list of functions available to be called by means of a simple *.bat file command see Appendix or S_TOOLS Reference Manual. STx can also be assigned to be associated with the file type *.wav, calling STx by double click from the MS Explorer file list. Full implementation of a DDE Interface is currently under development. By now, a Visual Basic Word Macro can be provided which enables the user to control play and viewer functions from Microsoft Word. For further information please contact the STx programmers team at toni@kfs.oeaw.ac.at.

S_TOOLS-STx USER'S GUIDE
What is STx? – Intelligent Sound Processing

1.1.3 Macro Programming
STx can be programmed using an easy-to-learn macro language. You can type commands and expressions directly into a macro file. This file is read by STx and the requested functions are executed. In this way, STx can be used in batch mode, e.g. for

- automated sound analysis procedures
- interactive experiments
- test designs
- automated signal and process control.

For a list of available functions and commands, along with the complete syntax of the macro language see the STOOLS Reference Manual. The standard macros which control most of the functions of STx are provided in source code, so that you can use them as templates.

1.1.4 Compatibility
STx creates, imports and exports MS Windows WAVE files. This enables the user to read and write soundfiles on external computer systems, and to exchange sound data in a network environment. Beware, however, that some non-standard sampling frequencies and soundfile formats created by non-STx systems may be incompatible or may lead to inaccurate results. Usually soundfile format converters can be applied in order to establish format compatibility.

The outstanding feature of STx is that it does not manipulate a whole WAVE file at time. Rather, STx accesses individual soundsegments in a soundfile by means of SOUNDSEGMENT DIRECTORIES. Standard WAVE file players only provide access to soundfiles as a whole—which can be very time-consuming with large soundfiles!

1.1.5 Sound Presentation System
Its archive functions make STx an effective tool for the development of highly interactive presentations (which can use very large sound databases). Since the main functions of STx are controlled by the STx macro language, the user can implement convenient user interfaces for interactive and educational applications. Graphics, text and sound can easily be combined in this kind of presentation. STx also supports the production of CD-ROM's containing such presentations and the relevant sound, graphic and text files.

1.1.6 Local and Distributed Sound Archive Applications
STx provides tools for local and remote data acquisition and for the management of mass storage (of sound and associated metadata) on RAID systems in a LAN, or in an Internet environment. Each soundfile is associated with a SOUNDSEGMENT DIRECTORY file. This
directory file contains information on the soundsegments in the soundfile (along with additional metadata), plus links to further documents stored on local or remote disks.

The SOUNDSEGMENT DIRECTORIES are designed to be used as input to standard database programs, hypertext systems etc. This enables the user to

- implement sophisticated data models by free definition and manipulation of the data
- safely store of a large amount of data
- freely control access to the soundfiles and associated documents.

All of these are properties of a comprehensive database management system.

Hint: Practical experience has shown that soundfiles should not be held longer than 5-8 weeks on local disks or on backup systems. For long-term storage, all documents should be placed in a professional archive system. Currently, migration periods of approximately 5 years are feasible for large-scale sound database and archive systems.

1.1.7 LAN Implementation

STx is fully compatible with standard network environments. STx communicates with other acoustics workstations, with local servers and with remote archive systems. In a typical LAN environment, 3 to 4 STx workstations can access sounds and metadata documents from a single server simultaneously. At sampling rates exceeding 48 kHz stereo, more powerful network connections are advisable in order to guarantee continuous data streams to and from multiple workstations.

1.1.8 Internet Service

STx provides an interface to widely used Internet audio and multimedia servers. Currently, software support is provided for the Real Audio server. A small-scale demonstration system of a digital sound archive providing access by the Internet has been set up under http://www.kfs.oeaw.ac.at/DLI/home.html. Comments and suggestions are much appreciated.
1.2 Installing STx for Windows

1.2.1 System Requirements

In order to fully utilise all the features of STx, the following minimum system configuration is suggested:

- IBM PC-compatible computer with a Pentium III processor 600 MHz or higher, running MS Windows 98/2000/NT.
- Accelerated Graphics Port (AGP) graphics card
- Standard sound card
- 128 MByte random-access memory (RAM)
- 8 GByte free disk space

STx itself requires negligible disk space. For storing sound data, no hard-and-fast recommendation can be given; but we would warn against underestimating your needs! We have found 30 GBytes to be convenient. As a rule of thumb: 700 MBytes is necessary for 1 hour’s worth of sound (HIFI quality, 48 kHz, stereo, 16 bit).

1.2.2 PC Audio System Requirements

STx supports standard sound cards (e.g. SB Live, analogue/digital or compatible) and professional sound cards, provided that Windows 98/2000/NT driver software is available. Professional cards must have the correct driver(s) because of their higher demands on system resources: they provide higher sampling rates and word lengths (96 kHz, 24 bit) as well as more than 2-channel analogue and/or digital inputs/outputs.

1.2.3 Program Installation

Note: If you already have an older STx system installed, please copy your old STx installation to a different disk or folder in order to temporarily save your personalised configuration files. After saving your files, delete the old C:\OEAW\SF\STX folder, then install the new version of STx. If you do not, the new installation will partly replace your existing STx program files, producing an STx version mismatch and the proverbial unpredictable results.

STx for Windows installation files are delivered in a compressed format on the original S_TOOLS disks. The self-extracting file setupyyyy.exe automatically creates the C:\OEAW-SF\STX folder on your computer. The setup procedure prompts the user to enter a valid license number (which is provided to registered users). The files are copied to your STx folder and an STx icon is created on your desktop.
To uninstall STx, please use standard Windows uninstall procedures, otherwise the system registry entries will not be deleted and you may have trouble if you ever re-install STx on that computer.

1.2.4 Online Help and How to Contact the STx Team

Regular updates of STx are available from the STx ftp server using your licence number as user-id and password:

ftp://xxxx-yyyy:zzzz@ftp.kfs.oeaw.ac.at

If you need technical support please do not hesitate to contact us by email: wad@kfs.oeaw.ac.at or by fax: +43-1-4277-9296.

1.3 Principles of Digitisation and Soundfiles

In order to digitise analogue signals properly – which turns out to be a lengthy process – a few preliminaries should be considered. Whenever an analogue signal is to be digitised, the process should be planned and executed with a mindset not to do so twice. This mindset implies:

- Performing analogue to digital (A/D) conversion and digital to analogue (D/A) conversion at the highest sample rate appropriate to the nature and the information content of the originals
- Performing analogue to digital conversion at the highest resolution and sampling rate appropriate to the dynamic range and sound quality of the originals. This avoids re-transferring and re-handling the originals in the future
- Creating and storing a linear-coded master soundfile that can be used to produce derivative filtered and/or compressed or otherwise processed soundfiles, in order to serve a range of current and future needs
- Creating backup copies, on a stable medium, of all soundfiles that are created.
- Creating meaningful metadata for soundfiles and associated documents, including (if required) cataloguing information according to a scheme that has been thought through ahead of time
- Monitoring the conversion and recopying data if necessary
- Outlining a migration strategy for transferring data to alternative sites, including the next generations of file servers.

STx currently works with sample rates selectable from 500 Hz to 100 kHz at word lengths of 8 or 16 bits, provided that these formats are supported by the driver of your sound card under Windows 98/2000/NT. A binary word length of 24 bits and alternative soundfile formats are available on request. For further hints about performing digitisation see the Sections RECORD and General Soundfile Setup.
1.4 Soundfile and Soundsegment Management: Basic Concepts

This section describes some basic concepts, techniques and methods for working with soundfiles, sound segmentation and segment management tools. The inviolable principle of STx is that a soundfile is never altered in any way during segmentation, annotation, analysis, display or manipulation. (In fact it can and should be a read-only file.) The only exceptions are:

- when creating (i.e. recording or exporting a new soundfile), or
- when deliberately deleting a soundfile (in which case you get a very noticeable warning message), or
- (depending on settings) when explicitly processing a file (see §0 below)

STx organises sound, associated data and signal processing tasks in an open multilevel management system:

1. Soundfiles
2. Soundsegments (audio segments)
3. Soundsegment directory files
4. Soundsegment lists and attributes

Figure 1: The multi-layer segmentation system for soundfiles. Sound segment addresses, segment identifiers, optional links and content descriptions (annotations) are stored in a separate reference data file called a SOUNDSEGMENT DIRECTORY file, normally located in the same folder as the current soundfile.
1.4.1 Soundfiles

Soundfiles are binary files containing streams of continuous sound data. They are organised on hard disks, in folders and subfolders, as standard *.wav files. Soundfiles primarily contain audio signal samples. Standard multimedia players provide analogue output of WAV files through the sound card’s Digital to Analogue Converter by reading the files as unique and contiguous streams of data.

STx allows sound to be accessed selectively, using

- fine-grained sound segmentation
- sound segment management
- sample-by-sample signal addressing

Additionally, extended user access to technical metadata is supported, e.g. to sampling frequency, sample coding as well as links to other documents associated with the sound. Simple metadata and links to files carrying ancillary information are stored in so-called User Chunks. For further information see: EBU Broadcast Wave Format or http://www.tnt.uni-hannover.de/soft/audio/tools/ff_convert

1.4.2 Soundsegments (Audio Segments)

A soundsegment (audio segment) is a selected and named sequence of contiguous sound samples in a signal stream. Segments are located in soundfiles. Access to soundsegments is provided by means of references in the SOUNDSEGMENT DIRECTORY that is associated to each soundfile and stored in a soundsegment directory file. So, for example, musound1.wav contains the sound data and musound.sd0 contains all metadata on sound segments in the soundfile musound1.wav.

![Figure 2: Soundsegment directory. From left to right: SegmentName StartTime Duration Annotation (optional).](image)
1.4.3 Soundsegment Directory Files

SOUNDSEGMENT DIRECTORY files are metadata files created automatically by STx on the same disk drive/path of a valid soundfile the moment that soundfile is created or first accessed in STx. Whenever a soundfile is processed in STx, soundsegment information and all other user-defined metadata (attribute definitions) are continuously updated and stored in the SOUNDSEGMENT DIRECTORY file associated with that soundfile. In other words, SOUNDSEGMENT DIRECTORY files are reference files to soundfiles; they are coded in ASCII and are automatically assigned a name comprising the filename of the current soundfile with a user-selectable extension (by default: *.sd0). The data contained in a SOUNDSEGMENT DIRECTORY file can be edited and exported by means of standard text editors. STx never alters the audio content of a soundfile except when you specifically ask it to (see §0 below); it only updates the content of the associated soundsegment directory.

1.4.4 Soundsegment List

Segment lists (tables) specify segment addresses (time pointers) and associated metadata (attributes and annotations). They are stored in SOUNDSEGMENT DIRECTORY files. They support the management of named soundsegments of continuous stored signals, whether located in the associated soundfile or in another soundfile.

1.4.5 Soundsegment (Audio Segment) Definition

A soundsegment is defined by a SegmentName, SegmentStartAddress and Segment Duration within a soundfile. However, soundsegment data are never stored in the soundfile itself, but only in the associated soundsegment directory file associated with that soundfile. For further information on absolute and relative sound data addressing and access in a soundfile see section 9.2.1 below (Soundsegment Addressing throughout All STx Modules).
2. **STx Overview**

2.1 **The STx Work Areas**

This and the following sections can be used as a hands-on walk-through of the basic features of STx.

2.1.1 **Startup**

When you start STx, the STx main application window (which is very small) appears in the left upper corner of the Windows screen:

![STx main window](image)

*Figure 3: The STx main window. At first startup of STx it appears in the upper left corner of your screen. Below it, the empty SOUNDFILE LIST window appears.*

By default, the `S_TOOLS` application is selected. So you will also see a second window:

![S_TOOLS SOUNDFILE LIST (SFList) window](image)

*Figure 4: The empty S_TOOLS SOUNDFILE LIST (SFList) window at first startup of STx*
Beware: STx provides a maximum of flexibility in parameter settings of various functions such as file naming, file formats, analysis procedures and display options. Later, you will appreciate this flexibility; but for the purposes of initial orientation, we encourage you to keep the default parameter settings until you have mastered the basic workings of the system. With only a few exceptions (such as selection of sampling frequency, soundfile format and soundsegment directory specification), almost all parameters can be altered later on without loss of data acquired using the default settings.

The next two subsections show you the most important functions of these two windows.

2.1.2 The STx Main Application Window

An STx workstation is controlled at the highest level by the STx main application window. The functionality of STx has been implemented in various tools/applications (i.e. groups of related functions) which can be started from this main window:

- **S_TOOLS**: used for the analysis, display and processing of soundfiles (with integrated sound-recording and file-search capabilities)
- **RT-Analyser**: a two-channel real time (RT) spectrum analyser for analog or digital input
- **Find Files**: a tool for locating soundfiles and sound segments on mapped disks
- **Sound Recorder**: records analog or digital signals from the currently activated input of the computer’s sound system.
- **Setup Manager**: provides direct access to the settings of STx for advanced users
- **Command-Line**: interface to the main application shell for advanced application development.

![Figure 5: The tools/applications available from the STx main window. By default, and at first startup, the S_TOOLS application is started, which opens the SOUNDFILE LIST window.](image)

Each of these applications can be selected from the run menu of the main STx window. The selected application starts in a window of its own, called a WORK AREA, and opens additional windows as needed. Though unusual at first, you will soon come to appreciate the flexibility which this allows you in displaying windows from different applications side by side.
The positions of windows are set to useful defaults, but you can move and re-size them as you wish, and STx will remember their position and size for the next time you open the same window.

The default application, which is opened automatically at the first startup of STx, is S_TOOLS, which is described in the next subsection. See section 0 below for alternative startup options.

### 2.1.3 The S_TOOLS Application – the Soundfile Work Area

The S_TOOLS application opens with the SOUNDFILE LIST (SFList) window (which contains no soundfiles at the first startup). This window displays a list of all the soundfiles available to the application.

Almost all soundfile and setup functions of STx can be invoked from the menu bar of this SOUNDFILE LIST window. The 5 pop-up menus provide access to the main control functions of STx.

![Image of STx soundfile work area](image)

*Figure 6: The STx soundfile work area as accessed through the SOUNDFILE LIST, and the 5 submenus which control it*

In addition, there are two useful buttons, record and find which save you the trouble of going through the STx main window to start these applications.

At the first startup after STx installation, soundfiles can immediately be opened or created on any of the disk drives (local or remote) available to the system (see the RECORD subsection below).
Once a soundfile has been opened or created, it appears in the soundfile list window. Soundsegments can be defined, then displayed, analysed and otherwise manipulated as described in the following sections. The various functions of the STx work areas are summarised in Figure 7.

2.2 Record – Start/Stop Recording

For the introductory part of this manual we will first look at the hard-disk recorder section of the STx system, and will treat it as if it were no more than a simple tape recorder.

Recording to a remote file server is only advisable if a continuous high data transfer rate can be guaranteed. For secure recording of original events, the use of a local drive is strongly recommended. Always ensure that you have enough free storage capacity and a high throughput.

The Record function of STx can be started either from the main window or from the S_TOOLS soundfile list window.
When Recorder is selected from the STx main window or Record from the soundfile list window, the recording section of STx is started. Signal capturing from analogue or digital sound inputs can be organised in a convenient cumulative procedure creating sequences of concatenated items. Single click Start-Stop-Start provides control of the live recording similar to analogue tape (Append Signal).

Any arbitrary Start and Stop time (or duration) of a recording can be specified if desired; this feature is useful if an earlier recording has to be repeated and parts or the whole of the recording has to be replaced by a new signal. Before beginning a recording, you should check the selection of sampling frequency, sample wordlength, number of channels and the disk folder where the temporary file is to be stored. We recommend that sufficient disk space for a continuous 3h recording session is always kept available.

**Important!** Before digitising, users should think very carefully about implementing appropriate file naming conventions for soundfiles, for soundsegments and associated documents. A content-related classification or indexing scheme used from the outset will permit effective data retrieval later on. As STx supports the management of a large quantity of soundfiles (up to several thousand), proper naming of files and well thought out folder structures have proven helpful in many applications.
2.2.1 Play/Record Setup

Although the default STx configuration for the sound record/reproduce function will work properly in almost all applications, STx provides the option of increasing and decreasing the number of wave buffers which the record/reproduce function uses. If you have two or more sound cards installed on your system, the setup menu allows you to select the sound card that is to be used.

Please note two important points:

- It is advisable to keep a carefully check on the configuration of your sound card(s) in the Windows system environment in order to guarantee the proper working of all record/reproduce functions. STx relies on the control panel “sound” settings and the current WAV IN/OUT configuration of the Windows sound card driver. We suggest that you keep the “sound” applet of the control panel open when playing or recording, in order to have direct control of the various settings. A mismatch of the sound system control settings can produce unpredictable results.

- Some disk drives exhibit uncontrolled self-calibration behaviour, which can cause buffer under-run. This is not acceptable during continuous disk recording. Avoid using such drives for real time sound recording.

![Figure 9: Mixer settings for analog or digital input (SPDIF-IN) of a Creative Soundblaster Live sound card under Windows.](image-url)

After the successful creation of a soundfile, its name and technical specification appear on the header of the soundsegment list and the soundsegment directory window is automatically opened. At this stage of creating a new soundfile the segment directory list is empty, with exception of the system-generated signal entries Signal.1st and Signal.All.
2.2.2 Input Signal Level Control (Metering)

For analogue inputs, the input volume control has to be carefully adjusted. Please remember that an analogue audio VU (Volume Units) meter indicates the average signal level. The actual peak level is often 10 or 15 dB higher. High quality analogue audio systems have a headroom of 20 dB up to +24 dBm.

A line input level electrical signal typically has a voltage ranging from 0.3 to 2 Volts, while a microphone level signal is more often in the range from 5 to 50 mV (millivolts). Microphone sensitivities range from -60 dBu to -22 dBu referenced to 94 dB Sound Pressure Level (0 dB SPL = 2*10^-5 N/m^2). The consumer line input level electrical signal typically has a voltage of 0.32 V (-7.8 dBu), whilst the professional line input level is typically 1.23 V (+4 dBu). For further information see: 9.8.1 Commonly used Voltage and Audio Levels.

Note: With consumer and semi-professional audio equipment, a VU reading of 0 dB is typically referenced to -10 dBV, which itself is referenced to 0 dBV = 1 V RMS (see section 9.8 below). Professional audio equipment works at a considerably higher level: a 0 VU reading corresponds to +4 dBu. Connecting a professional +4 dBu device to a consumer audio input may produce dangerous overloading, whereas the output of a consumer device probably does not have sufficient power to drive a professional audio input.

Digital systems have no headroom at all. Applying a signal beyond the maximum input level results in severe distortion, so-called digital clipping, which has to be absolutely avoided. A clipped signal will almost always sound unnatural; it will most likely sound horrible, not just slightly distorted. As best practice peak level adjustment between -18 to -12 dB full scale (full code maximum input level), typically at -15 dB, is suggested. Peak levels are measured by means of so-called peak meters, indicating absolute positive or negative signal peaks. Full scale resolution is in direct relation to sample resolution or
Creating Tags and Cue Points during Recording

Sample depth (bits per sample) and the dynamic range covering the audio signal’s full amplitude variation over time. Currently, the most common resolution on PC systems is 16 bits (linear PCM). For high-end audio applications 18, 20, 24 or even 32 bits per sample are appropriate.

Modern sound cards provide digital inputs alternatively. Digital signals captured via digital audio streams by means of the professional AES/EBU interface or the widely compatible consumer version S/PDIF generally need no level control. It has to be assumed that level control has been performed in the course of the first digitisation process. To probe further see: Digital Audio Interface Standard IEC958 from EBU (European Broadcasting Union).

2.2.3 Dynamic Range

Dynamic range is defined as the ratio of the full scale (FS) signal level expressed in dB FS to the RMS noise floor, in the presence of a signal. This specification is frequently referred to as Signal-to-Noise Ratio (SNR). Dynamic range is usually tested by measuring the total harmonic distortion plus noise (THD+N) with a -60 dB FS signal. A theoretically optimal system provides the full scale range as its dynamic range. Typical values of PC-Audio D-A systems are measured at the output with a dynamic range of 85 dB FS A, where the “A” suffix indicates that an A-weighting filter has been applied. Professional studio record equipment today provides a dynamic range of 107 dB FS A and more, at sample word lengths of 18 to 20 bits.

Table 1. Full Scale Range (dB FS) available at different digital word lengths (bits / sample).

<table>
<thead>
<tr>
<th>coding</th>
<th>bits/sample: n</th>
<th>8</th>
<th>16</th>
<th>18</th>
<th>20</th>
<th>22</th>
<th>24</th>
<th>32</th>
</tr>
</thead>
<tbody>
<tr>
<td>FS ratio: N/1</td>
<td>2^8</td>
<td>2^16</td>
<td>2^18</td>
<td>2^20</td>
<td>2^22</td>
<td>2^24</td>
<td>2^32</td>
<td></td>
</tr>
<tr>
<td>FS ratio: dB</td>
<td>48</td>
<td>96</td>
<td>108</td>
<td>120</td>
<td>132</td>
<td>144</td>
<td>192</td>
<td></td>
</tr>
</tbody>
</table>

Note: dB FS = 20 * \(\lg(N) = 20 * \lg(2^n)\)


2.3 Creating Tags and Cue Points during Recording

The STx Recorder provides the option to create tags and segment addresses already during the initial recording of a sound. The level control of the recorder is designed to display the waveform stepwise in periods of 1 second. The user is enabled to insert cue points at arbitrary positions of the waveform envelope currently displayed. By hitting [Shift+left mouse button] a tag is created at the momentary location of the mouse cursor which can be used as a cue point for further segment processing.
2.4 **Play – Reproduce**

**Play** starts the audio reproduction of sound stored in a soundfile. Usually **Play** is started by means of a double mouse click on a soundsegment item listed in the segment directory window of a soundfile.

Many signal view functions, as well as the spectrogram display, (sectioner) provide the hot key `p` for starting the play function for that part of the signal that is currently bracketed by two cursors. If multiple segments have been selected, the segments will be played in the order of their appearance in the soundsegment directory. In order to play sequences of segments from different soundfiles in one stream, the Sequencer function of STx can be used.

**Play**... needs the start and end address (or, alternatively, duration ) of a reproduce function to be stated explicitly. (**Play** 600s_+300s reproduces the sound beginning at absolute file time 10 minutes with a duration of 5 minutes (10′-15′)).
Play – Reproduce

You will need to have several soundfiles open simultaneously in order to conveniently process a number of soundsegments located in different soundfiles. Usually the original recordings, which are stored on individual soundfiles, are not in the configuration or sequence in which you wish to listen to or to process them. This is why STx provides free absolute and relative sound addressing, sample by sample, on any soundsegment and in any soundfile placed in the STx work area (see Appendix: Sound Segment definition).

This is the end of our first overview of STx. The more advanced items from the File and Run menus of the STx main window, which were not dealt with in this chapter, are treated in sections 0 and 8 below.
3. Soundfiles and Sound-Segments in STx

In this chapter you will find all the details of soundfile management in STx and of sound-segment definition and manipulation. We begin with the powerful Findfile application, which will greatly facilitate the integration and management of your soundfiles, no matter where they are located in your system.

3.1 Findfile – Add Soundfiles to the Work Area

This application finds all files that meet your search criteria in all sub-folders of the selected folder(s) or drive(s), on both local and mapped network disks. The standard setting is to search for soundfiles (indicated by soundfile search in the top left corner of the Findfile window).
Figure 13: Find soundfiles on attached disk drives and Add them to the work area of the current STx session. The pop-up window is opened by clicking the right mouse button inside the list pane.

To search for sound files, you need to:

- open the **Findfile** application, either by a single left click on the files button in the soundfile list, or by selecting **run – findfiles** from the STx main window
- select the drive or folder to search in, using the **add** or **remove** buttons. (The **add** button allows you to select the drive (top pane) and folder (bottom pane), and to choose whether or not to search in sub-folders (**sub-dirs**)..) You can add more than one location for the same search.
- click on **< Start >** button

You will notice that after STx has found all the soundfiles, it checks their integrity before displaying them in the bottom pane of the **Findfile** window. (This checking can be disabled by de-selecting the **soundfile-check** box).

Now you can select files and either add them to the soundfile list (right-click pop-up menu item **add to workspace**) or open them immediately (pop-up menu item **open**), simultaneously adding them to the soundfile list. To open just a single file, you can double click on it. To get the pop-up menu, click on the right mouse button when the cursor is within the file list pane of **Findfile**.

**Beware:** we recommend that you do not open more than 8 soundfiles simultaneously, so that your computer does not run out of Windows resources.
Once a file is added to the soundfile list, a soundsegment directory file (*.sd0) is automatically created in the folder where the soundfile is located.

Notice the other practical options in the pop-up menu of the found files list. These should be self-explanatory, so only the summaries will be mentioned here.

### 3.1.1 Folders/Files Summary

These two types of summary (from the right-click pop-up menu of Findfile) provide useful information on the folder that is selected for soundfile search or on the selected soundfiles. This information includes the number and disk locations of soundfiles, the cumulative duration of total sound and disk space occupied.

### 3.1.2 Advanced Findfile Options

In the Findfile window you will see various further options which will turn out to be very useful in managing large soundfile databases.

**Hint:** Try expanding the Findfile window horizontally and scrolling horizontally to the right. Here you will find all kinds of useful data on each soundfile.

The file list in the Findfile window can be sorted according to the primary, secondary, etc. criteria given in each of the 5 sort selection boxes (from sort 1st to sort 5th). Each sort criterion can be switched from ascending to descending order by checking the reverse box. The List Settings button allows you to select or de-select the categories of information displayed in the file list. These are:

- directory (path):
  displays the name of the folder which contains the soundfile(s)
- file size:
  displays the size of the soundfile in Bytes
- modification date:
  displays the date on which the soundfile was last modified
- modification time:
  displays the time at which the soundfile was last modified
- sampling rate:
  displays the sampling frequency at which the soundfile was digitised.
- number of channels:
  displays the number of channels in the soundfile (1ch = mono, 2ch = stereo)
- signal length:
  displays the length of the signal contained in the soundfile (in seconds, with 2 decimal places)
The cluster of buttons at the top right of the Findfile window allow you to save and load file lists to a list file of your choice.

Finally, let us return to the top left button and its associated text entry field. When Findfile is first opened, the button indicates soundfile search. If the text field to the right of the button is left blank, the program assumes that the filename is the “star” wildcard (*) and finds all soundfiles. But you can restrict the filenames by entering alphanumeric characters or wildcards in this field. Thus, for example the entry music* will find all soundfiles whose filename begins with the string music, e.g. music_traditional_Irish.wav, etc. You can also use the question mark wildcard (?) in the search mask, in place of one character of the filename. So a search for 2000-??-01 will find files like 2000-01-01.wav, 2000-12-01.wav, 2000-ab-01.wav etc.

If you click on the soundfile search button, you are offered different types of searches in sequence. (So to get back to soundfile search, you just keep clicking on the button until those words reappear.) Here is a synopsis of the functions of the other types of file search:

- **same name**
  finds pairs (3’s, 4’s etc.) of files which have the same filename and file type in different sub-folders of the folder/disk that is searched. If the search mask field is left blank, *.* is assumed.
- **same name+size**
  performs the same type of search as same name, except that the size of the files must be identical, too.
- **general search**
  finds files of any type (not just soundfiles) as specified in the search mask field. If the field is blank, *.* is assumed, and all files in the selected location (folder or disk) are listed.

The Findfile application has been tested on a data corpus of more than 2000 soundfiles stored on a 100 GByte RAID server. A soundfile search of the entire volume created a list of accessible soundfiles in less than 10 minutes. The correctness of the list was checked manually and fully verified.

**Remember:** A double click in the Findfile file list opens the file. A right click gives you the powerful pop-up menu.

### 3.2 The Soundfile List Window

The soundfile list (S_Tools SFList) window is the main control centre of the interactive part of the STx work area. By now you should be familiar with the procedures for selectively adding soundfiles to the soundfile list. So here we deal with other aspects of soundfile lists.
3.2.1 Remove a Soundfile from the Work Area

This function removes one or all selected soundfiles from the current work area (soundfile list). On removal, its soundsegment directory window, if open, disappears from the screen. Removing a soundfile from the list is non-destructive for the soundfile located on the disk. The content of the soundfile and that of the segment directory file itself are not affected by removing the file from the soundfile list.

*Hint:* If you accidentally remove a file from your soundfile list, the find button is the quickest way to re-locate it and add it again.

3.2.2 Automatic Saving of the Work Area

When you close STx by means of Exit (from the File menu of the main window), the current default soundfile list will be saved and all soundfiles which are open at the end of the current session will automatically be opened the next time the program starts. This allows you to continue your work in multiple sessions without having to remember which file(s) you were working on.

If you wish, you can disable this feature in the Setup – Options dialogue box by deselecting the checkbox remember opened soundfiles.

3.2.3 Delete a Soundfile from Disk

Deletes all selected Soundfiles: deletes the soundfile and its entire signal content from the disk. **Be warned – this function is destructive!** The soundfile and the accompanying soundsegment directory file are deleted. If you want to keep the directory file whilst deleting the soundfile, the directory file must be renamed before executing the delete function. Currently no UNDO function is available. Deleted files are not placed in the Windows recycle bin!

3.2.4 Save File List (As) - Load File List

With this function you can:

- **Save** the content of the current soundfile list in an ASCII text file.
- **Load** the content of the soundfile list from a specified ASCII text file.

It is advisable to store a properly edited soundfile list of your current STx session on a disk file for later reconstruction of the same work area. It is intended that users save their personal work area settings and system configuration on specified files in order to support cumulative work in different sessions or by more than one user. Otherwise there is no guarantee that a user will find his previously defined work area as the default work area on the same workstation if it has been used by someone else in the meantime.
3.3 Sound Segment Labelling and Metadata Management

We now shift from the external management of soundfiles to their internal management and manipulation. The key concept here is that of a “SOUNDSEGMENT”, described in the next subsection. Please read the whole of this section before deciding which method(s) of defining and manipulating soundsegments best suits your purpose.

3.3.1 What Are Soundsegments?

A soundsegment is any part of the signal, from zero length to the full length of the soundfile. Each soundsegment has a unique name within a soundsegment directory file. You can define soundsegments in a number of different ways, and can customise your segment template to incorporate precisely the information you want to associate with your soundsegments.

Customisation of the segment template is carried out via the Setup menu in the soundfile list window. From this menu, select Segment Template, and the Segment Template Management window opens. Here you can modify the contents of the default template, then save it under a different name. For an explanation of the options available for each entry in the template.

*Hint:* You will probably find that the default template serves your immediate needs until you are familiar with the workings of STx. Modifications of the segment template should on the one hand be well thought out, but on the other hand be finalised before large amounts of sound data are segmented.

All soundsegment directory information is stored in ASCII text files with extensions *.SD0, *.SD1,... SDx, each of which is located in the same disk folder as the soundfile to which it refers. Soundsegment directory files can be edited with standard text editors and provide convenient access to soundsegment addresses, names and annotation data by means of any database management system. Soundsegments are managed and accessed by means of the Sort, New, Modify and Delete functions from the Segments menu of the Soundsegment Directory Window of any soundfile (as shown in Figure 15 below).
3.3.2 How to Specify and Manipulate Soundsegments

Soundsegments can be created in arbitrary order, even overlapping in the same soundfile as well as within already created soundsegments. Segment addresses are allowed to be written in simple numerical expressions and combined syntax such as `Name.Ext+xs_ys-z`.

*Note:* the underline character (_) defines the from_to signal address pointer syntax in order to reserve +/- exclusively for use as symbolic signs in numerical expressions.

Additionally, the Beginning of a segment (:B), the End of a segment (:E) and the Duration of a segment (:D) can be used as identifiers for address pointers and values. Soundsegment addresses can be specified in samples by means of sample numbers (e.g. 44100+44100) or time pointers in s (e.g. 120s) or ms (e.g. 3000ms) relative to the signal start 0 of the current soundfile.

For further description of the syntax of STx segment addresses and address expressions (for the execution of commands such as Play..., View..., Analyse... etc.), see section 9 below.
3.3.3 Segments – New: Creating an Entry for a Segment (Name)

New generates an new entry in the soundsegment directory. The Name or label of the segment is usually entered by the user. The beginning and length (duration) of the segment are entered in the Begin/Task and Length fields respectively. These values are automatically pasted from the automatic generated address pointer values of the last executed record or signal processing function (in adc.tmp), but can of course be modified by the user. For the narrow segmentation of large quantities of sound signals (phonetic transcription of speech), an automated labelling and name function is available, which speeds up processing considerably.

3.3.3.1 Segment Annotation and Attributes

The Type, Text and Tag fields are examples of user-defined fields for convenient annotation and description of soundsegments. The definition of such descriptive fields is carried out in the soundfile template menu (Setup – Segment Template – Edit in the soundfile list window). You can create different types of metadata structure by defining the appropriate fields in the corresponding templates. Further support, in the form of automatic segmentation, naming and annotation procedures for several classes of acoustic signals are under development.

3.3.3.2 Segments – Modify: Changing the Entry of a Segment

Modify displays all the current data of a segment (i.e. its name, addresses and annotation data) and allows you to modify the content of any of the fields. Rename enables you to change the entry Name of a segment.

Note: users should think very carefully about implementing appropriate naming conventions for soundsegments, annotations and associated documents. A content-related classification or indexing scheme used from the outset will permit effective data retrieval later on.

3.3.3.3 Segments – Sort: Sort the Entries in a Segment List

Sort allows you to specify the order in which the list of entries in a soundsegment directory list is presented. Typical sort functions include standard fields like SegmentName – Begin – Duration – Type – Text – Tag (ascending or descending order; invers indicates descending order). The sort mask supports up to 3 sort criteria. Multiple sort levels can be achieved by successive processing (e.g. with 6 sort criteria, sort by the 4th, 5th and 6th criteria, then the 1st, 2nd and 3rd).

STx supports the totally free creation of field names, and so a large number of fields (numeric and/or text) can be added for each soundfile category. Field contents for annotation and metadata generation are used to perform highly complex reorderings. This
function is therefore a convenient tool for the creation of the appropriate listings for multiple soundsegment processing.

3.3.4 Segments – Delete: Erase a Segment

Delete erases the SegmentName and all associated data of the selected segment. If selected, this function is carried out immediately. **Beware: this function is destructive! Currently no undelete function available!** The old soundsegment directory file is **not** placed in the recycle bin.

3.3.5 Segments – Sequence: Open the STx Sequencer

Sequence starts the sound sequencer application of STx and allows you to add the currently selected soundsegment to a new or existing sequence. Once the sequencer is opened, you can freely modify sequences.

3.4 Taking Care of Your Soundsegments

STx uses the continuously updated soundfile directory files as reference and control files for all signal-dependent functions which refer to soundsegment and sound signal addresses. Directory files and their metadata content are the main input to the Database Management System.

**Beware:** If soundfiles are moved or copied from one location to another, you need to make sure that all associated soundsegment directory files (*.sd?) are also moved/copied. Recall that STx expects the directory file to be located by default in the same folder as the soundfile.

In addition to the soundfile directory itself, all directly associated data and links to a soundsegment are stored in the soundfile directory file. This file is written in ASCII format and located in the same disk folder where the soundfile is usually stored. The extension of the Soundfile-Directory File is specified as *.sd0 by default. If more than one soundfile directory is associated to a single soundfile, the filename extensions of further directory files...
files are specified as *.SD1, *.SDx etc. Some signal analysis and signal processing procedures may require specific segmentation strategies and different segment directories.

**Hint:** It is advisable to create sound segment names (labels), tags, annotations, catalogue entries and additional metadata on the fly or immediately after recording. Naming should follow a previously agreed system relating to recording protocols, transcriptions, links, etc.

Although standardisation of metadata can be performed after recording sessions have been completed, sophisticated tagging of soundsegments and metadata information should be carefully planned in advance in order to facilitate the integration of text, graphics and sound in multimedia database systems.

### 3.5 Soundsegment Database (in preparation)

The soundsegment database function of STx (which is accessed via Tools → Database from the soundfile list window) is currently under development and therefore implemented only in a rudimentary form. It is intended to automatically find and list soundsegments with particular properties within a soundfile list. The resulting lists of soundsegments, along with their related metadata, can then be saved and later used for automatic processing in macros.

Since this function is likely to change radically in future versions of STx, it is not described here.

![Figure 18: Soundsegment Database: provides access to soundsegments and referenced metadata stored in selected soundfiles and associated soundsegment directory files.](image-url)
4. **STx Waveform Edit and Time-Domain View**

### 4.1 Preliminaries

Almost all STx functions relating to time-domain viewing and to the editing, displaying and processing of signals are highly configurable. Any combination of signal display modules, graphics dimensions and layouts, variables and parameters can be configured and used to meet the user's requirements. Nevertheless, all users are advised to start with the default configuration, and changes should be carried out incrementally at first, to avoid unexpected results. Experienced users may find the standard STx macros useful as templates for creating new interactive menus and menu items. An STx system can thus be completely redesigned and tailored to the user's needs, in order to obtain optimal results.

### 4.2 Waveform Display and Analysis (Viewer1)

#### 4.2.1 Built-in Display Types

The most basic types of waveform display are activated via the View and Analyse functions. To display a soundsegment, you need to

- open a soundfile from the soundfile list or Findfile list window (either by a double click on the filename or, to open multiple files, by selecting them, right-clicking in the list pane and choosing Open from the pop-up menu)
- select a soundsegment from the list which appears when you open a soundfile
- choose View or Analyse from the Signal menu or from the right-click pop-up menu
- select the type of view or analysis that is to be displayed
If you selected View, the viewer window opens immediately. If you selected Analyse, you still have to select one of the 9 types of analysis available from the Analyse pop-up window and click on the Start button.

For the Analyse part of this window, please consult chapter 5 on page 43 below after you have familiarised yourself with the viewing functions described in this chapter.

Alternatively (especially when you have a large number of soundsegments and know their names) you can choose View... or Analyse... from the Signal menu and enter a segment name manually.

The procedure for viewing or analysing a soundsegment is shown schematically in 4 steps in Figure 19.

The default soundsegment Signal.All, which is automatically generated by STx when a soundfile is added to the soundfile list, provides you with an easy way to view or analyse the whole soundfile as soon as you open it, even before you have defined soundsegments of your own.

![Figure 19: Four steps for analysing and displaying a soundsegment: [1] select soundfile from soundfile list window (here: Stockholm.wav), [2] select soundsegment from the soundsegment list, [3] select Analyse from the Signal menu [4] select a specific analysis type and start.](image)

The STx waveform display can be zoomed to a specified magnification. It is also easy to zoom along the vertical axis of a waveform using hotkeys (see section 4.3.1 below and...
the full reference chart in Table 17 on page 87). Specific waveform tools are available in a
number of screen areas and through associated hotkeys.

4.2.2 Custom Displays and Analyses
Several STx-users have already created their own Display- and Analyser-macros in order
to meet their special needs. As interactivity is crucial in many applications specially
designed and timely tuned user interfaces can be implemented by macro programming.

4.2.3 Signal Addressing
In order to address a soundsegment as input for any of the View or Analyse functions, a
signal segment can be selected by highlighting a specific soundsegment directory
entry(ies), as described above. Alternatively standard or blocked mode addressing can be
selected when using one of the “...” menu items from the Signal menu in the sound-
segment list window. Standard signal addresses can be entered in [samples], [ms] or
[s], or by means of name expressions as described in section 9 on page 75. (Note that
such addresses do not need to have been defined as soundsegments.)

Blocked addressing is useful if large signal segments have to be partitioned into
subsegments of equal duration (block length). When blocked addressing is selected,
segments of long duration, e.g. 3600s, can be partitioned into smaller, e.g. 1 minute
subsegments, simply by specifying either the block length or the number of blocks. STx
will be forced to process the sequence of subsegments sequentially.

4.3 Waveform Viewer1 Layout
The signal viewer/edit section of STx provides multiple display and segmentation options
for any signal under scrutiny. (Recall that it is opened with Signal – Analyse from the
soundsegment list window.) When compared with common waveform display methods,
STx Viewer1 does not provide a gliding window during playback of an audio signal.
Instead, gliding waveform and spectrogram display is provided by the STx Real Time
Analyzer application. Viewer1 “freezes” the selected signal of a soundfile and offers
highly flexible signal bracketing, segmentation and zooming functions in order to guaran-
tee fast access to sound signals and convenient browsing in long-duration sound re-
cordings. Annotation, linking and tagging as well as browsing in a large collection of
annotated segments is a central feature of Waveform Viewer1. The Viewer1 layout in
Figure 20 (“3 Lines”) shows one of the many possible configurations (from top to bottom):

1. zoom windows, showing the local signal at the time cursor positions through a
   large magnifying glass (optionally you can have just one zoom window)
2. waveform traces (segment lines) containing the soundsegment selected from the
   file and bracketed by the cursors in the overview line.
3. overview line containing the complete signal in this soundfile, displayed as a waveform envelope (optionally it can be displayed as a ruler line).

If the segment that is viewed is shorter than the entire soundfile, the parts of the signal located outside the segment can only be accessed via the overview display trace (= the bottom line of Figure 20). The overview display trace always starts at the beginning of the soundfile (time 0.0) and will be aligned to the left edge of the display. The time value at the right end of the overview display will be determined by the actual length of the soundfile that is being viewed.

If the file is stereo, the Waveform Viewer1 can be configured to display channel-1, channel-2 or both channels by selecting all, 1st or 2nd in the channel selection box of the Analyse window. The standard playback functions are not affected by the Viewer1 mono/stereo setting; the stereo signal can be monitored in stereo whilst Viewer1 displays either channel.

4.3.1 Cursor Functions and Hotkeys

Several hotkeys and cursor functions support fast access and browsing through large soundfiles. Most waveform display and signal playback functions are cursor controlled.

4.3.2 Signal Playback

The Play (signal reproduction) function is conveniently available at almost all stages of an STx session. If a waveform, a spectrogram or a parameter display is active, provided that time cursors are available, hotkey [P] will reproduce the signal located between the time points currently marked by the cursors. A highlighted entry in the soundsegment directory is played with the hotkey [P] or a double mouse click. For further hotkeys for reproducing specified portions of soundfiles, marked segments and display windows see Table 2 below.
Waveform Viewer1 Layout

Figure 20: Analyse – Waveform 3 Lines: from top to bottom: [1] zoom windows for left/right cursor of the 3-line display (window length +/- 40ms each), [2] 3 waveform traces showing the total length of the segment take#04 (191s), [3] waveform envelope of the complete soundfile Neapel.wav (duration 3451s). Arrows point from the location of the segment take#04 (selected in the previous dialog box) to its beginning and end in the 3-line display.

Table 2: Waveform Viewer1 Hotkeys

<table>
<thead>
<tr>
<th>Hotkey</th>
<th>File Overview Window</th>
<th>Segment (Line) Window(s)</th>
<th>Zoom Window(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>[P]</td>
<td>play signal bracketed by cursors</td>
<td>2 cursors only1</td>
<td></td>
</tr>
<tr>
<td>[Q]</td>
<td>play soundfile</td>
<td>play soundsegment</td>
<td></td>
</tr>
<tr>
<td>[Space]</td>
<td>play signal window before – after active cursor position</td>
<td>play window</td>
<td></td>
</tr>
<tr>
<td>[Esc]</td>
<td>close active dialogs – stop play immediately</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[F4]</td>
<td>increase play window length (double length)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Ctrl-F4]</td>
<td>decrease play window length (half length)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1 Only if 2 zoom cursors are displayed (zoom mode = “Between Cursors”)
4.3.3 Cursor Navigation

Generally, each waveform display provides two cursors for convenient signal bracketing, defining the “temporarily active” signal region. By default, one of the cursors is “active”. The active cursor can be selected and moved by pressing and holding the left mouse button. A waveform display trace is activated by clicking somewhere within the display area.

The STx display cursor can be moved if it is activated when its horizontal/vertical axis value display comes up on screen next to the Windows cursor (showing the current time and amplitude value of the STx display cursor). Hotkey [F2] or a click on the second cursor activates the STx cursor.

Again, holding down the left mouse button on the position of the cursor lets the cursor travel to the left or right in order to browse through the parts of the signal that are displayed.

![Figure 21: Move the Windows system cursor close to the STx waveform cursor. Whenever the current x/y values (= current time and amplitude reading) are displayed above (top graphic) or below (bottom graphic) the Windows cursor position, the STx cursor (here purple) can be moved by clicking the left mouse button and dragging, or with cursor hotkeys (see Table 3) when the STx cursor is activated (by a left mouse click or F2/Ctrl-F2).](image)

In addition to standard mouse and keyboard cursor functions, and in order to support strictly incremental processing, the following function keys and hotkeys have been implemented:

- [F2] toggle left or right cursor to active cursor.
- [F3] / [Ctrl-F3] mirror the inactive cursor on the time position of the active cursor / mirror the active on the inactive without changing the distance between both. This function is used for incremental or decremental stepping through a soundfile without overlapping.
- [L] lock or unlock (toggle) the time distance between both cursors. Locking the distance enables the user to produce soundsegments of exactly the same duration.
- [C] changes the cursor type (mode) which can be selected from: vertical bar (time axis only) vertical bar plus cross only (frequently used for x/y axis readout). Cross type cursors can be moved freely over the display area. Alternatively hotkey [B] forces the cursor to follow the function values displayed.
Practical experience with soundfiles containing a large number of segments as used for fine-grained text description or phonetic transcriptions has shown the necessity of fast access to segmented portions of a soundfile. The following hotkeys support fast browsing through soundfiles on a segmental basis in the Waveform Viewer:

- [S] snaps to a marked segment begin boundary located next to the active cursor
- [X]/[Y] jumps to the next/previous marked segment
- [Ctrl-S] saves cursor positions, i.e. sets the begin and end addresses of the selected marked segment to current cursor positions.

The last three hotkeys conveniently allow the auditory checking of already marked segments along with fast address corrections.

Note: These segment-related functions are only available if the waveform viewer has been configured to show marked segment boundaries and segment names (see Waveform Viewer1 setup).

Table 3: Waveform Viewer1 Hotkeys – Cursors

<table>
<thead>
<tr>
<th>Hotkey</th>
<th>File-Overview Window</th>
<th>Segment (Line) Window(s)</th>
<th>Zoom Window(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>[←],[→]</td>
<td>move active cursor left/right</td>
<td>move active cursor left/right and wrap around line(s)</td>
<td>move active cursor left/right</td>
</tr>
<tr>
<td>Shift-[←],[→]</td>
<td>move active cursor left/right (in larger steps)</td>
<td>move active cursor left/right (in larger steps)</td>
<td>move active cursor left/right (in larger steps)</td>
</tr>
<tr>
<td>[↑],[↓]</td>
<td>(not applicable)</td>
<td>move active cursor to previous/next line</td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[Pos1],[End]</td>
<td>move active cursor to begin/end of file</td>
<td>move active cursor to begin/end of line</td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[Page Up]</td>
<td>(not applicable)</td>
<td>cursor to first line</td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[F2] (cyclic)</td>
<td>activate first/next cursor</td>
<td></td>
<td>2 cursors only1</td>
</tr>
<tr>
<td>[F3] (toggle)</td>
<td>mirror inactive cursor around active cursor position / back to its original position</td>
<td></td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[Ctrl-F3] (toggle)</td>
<td>mirror active cursor around inactive cursor position / back to its original position</td>
<td></td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[L] (toggle)</td>
<td>lock/unlock cursor distance</td>
<td>(not applicable)</td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[C] (cyclic)</td>
<td>change cursor mode</td>
<td>(not applicable)</td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[S]</td>
<td>(not applicable)</td>
<td>snap to marked segment next to active cursor</td>
<td>(not applicable)</td>
</tr>
</tbody>
</table>
4.3.4 Waveform Zoom Windows

STx provides waveform zoom windows in the Waveform Viewer1 as well as part of the so-called Sectioner in the spectrogram display. Waveform Viewer1 offers 4 modes of signal zooming controlled by one or both waveform cursors; the selection of a specific zoom window type is performed by pressing hotkey [Z] in cyclic mode:

1. mode “between”: show signal bracketed by both cursors
2. mode “around 1 left”: show signal centred at cursor 1
3. mode “around 1 right”: show signal centred at cursor 2
4. mode “around 2”: show two zoom windows, signal centred at cursor 1 & 2
Table 4: Waveform Viewer1 Hotkeys – Zoom Window(s)

<table>
<thead>
<tr>
<th>Hotkey</th>
<th>File Overview Window</th>
<th>Segment (Line) Window(s)</th>
<th>Zoom Window(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>[*, [ ]</td>
<td>increase/decrease zoom window length (double/halve x-axis)</td>
<td>(not applicable)</td>
<td>set cursor to function “bound” (i.e. both cursors move simultaneously)</td>
</tr>
<tr>
<td>[+], [–]</td>
<td>increase/decrease zoom window waveform (double/halve y-axis)</td>
<td>(not applicable)</td>
<td></td>
</tr>
<tr>
<td>[B] (toggle)</td>
<td>(not applicable)</td>
<td>paste zoom cursor positions left/right to segment window cursor positions 1st/2nd</td>
<td></td>
</tr>
<tr>
<td>[1], [2]</td>
<td>(not applicable)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Z]</td>
<td>zoom window: show signal between, around 1, around 2 cursors</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The original zoom window length is specified in the Viewers general setup. In order to change this setting online, click on the desired zoom window, press [*] to double and [?] to halve the current time basis (zoom in – zoom out) and [+ ] to double and [–] to halve the amplitude scale. In practice, narrow segmentation needs accurate positioning of the cursors at selected points of the waveform’s fine structure. For this reason, both of the zoom windows cursors, which by default are located at time zero in the middle of the zoom window, can be moved independently under auditory control in order to identify the exact cue-in and cue-out points of a segment. Hotkeys [1] and [2] update the time position in the left and right zoom window cursor to the segment waveform trace display.

![Figure 24: Waveform zoom windows: zoom windows of left/right cursor (window length +/- 40ms each). The cursors of both zoom windows, which by default are located at time zero in the middle of the zoom window, can be moved independently under auditory control in order to identify the exact cue-in and cue-out point of a segment. After adjustment, hotkeys [1] and [2] update the cursors in the waveform line display.](image)

4.3.5 Segment Naming, Tagging and Annotation

Segment naming, specification and annotation is performed by bracketing the signal selected and by pressing the [INS] key which invokes the interactive signal segment
naming and annotation dialog. The annotation window can alternatively be opened by selecting Name from the pop-up menu which appears when you click on the right mouse button. Further navigation tools are [↓] and [↑] for switching to different lines of the multi-trace waveform display.

Table 5: Waveform Viewer1 hotkeys – creating a soundsegment entry

<table>
<thead>
<tr>
<th>Hotkey</th>
<th>File Overview Window</th>
<th>Segment (Line) Window(s)</th>
<th>Zoom Window(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Ins]</td>
<td>create soundsegment bracketed by cursors</td>
<td>(not applicable)</td>
<td></td>
</tr>
</tbody>
</table>

Figure 25: [INS]: Segment Naming, Tagging and Annotation window. Default soundfile template. In order to create a new template see 9.1 Defining Soundsegments.

4.3.6 Viewer Cue Point Lists (currently inactive)

Each viewer creates its own cue points (activated by pressing the right mouse button, either in a zoom window or on a segment line). Cue points are saved and managed in the viewers list of cue points. The list is needed in order to define segments using cue points that are not displayed and referenced by means of cursors in the current viewer windows. This is often the case when long soundsegments have to be manipulated.

4.4 Configuration of Waveform Viewers

The Waveform Viewer is fully configurable. Analysis of a long signal may require as many as eight segment tracers, whilst a short signal may require heavy use of the zoom windows.

Multiple instances of Waveform Viewer1 can be used in order to process different signal segments either in the same or in different soundfiles.
Configuration of Waveform Viewers

Figure 26: Cascading multiple Waveform Viewer windows. In order to cascade multiple Viewer windows, select Cascade from the Process menu in the soundfile list window.

The Process menu in the soundfile list window provides several options for arranging the size and order of multiple viewer windows. A laptop, however, may require this to be set to 768*576 pixels. Up to 4 viewer windows are supposed to be tiled on one screen. Cascading can be performed by selecting Cascade from the Process menu. Although STx does not limit the number of viewers simultaneously active in a single session, limitations are given by the system resources available at the local installation. As a general rule, 6 to 8 windows open simultaneously should work properly.
5. **STx Spectrum Analysis**

### 5.1 STx Spectrum Viewer2

Natural sounds are rarely simply sine waves. They tend to have complex narrow-band spectra, fast varying broadband transients, and additional noise components. Nevertheless, they can still be fairly well represented by a superposition of sine waves. For this reason, the Fast Fourier Transform technique (FFT) is still the workhorse for most frequency analysis, although alternative spectrum estimation methods such as the autoregressive (AR) models, wavelet analysis and Wigner Distribution applications are becoming increasingly popular.

Sound spectrograms originally have been obtained as the output of a heterodyne filter analysis provided by the sound spectrograph. Today the squared absolute values of the Short-Time Fourier Transform (STFT) are used in order to compute the distribution of frequencies as a function of time. STx Viewer2 currently performs the following spectrum analysis and signal processing tasks:

- Frequency analysis (short-term spectrum, spectral averaging)
- Spectrograms for the visualisation of time varying signals
- Wigner distribution and other time-frequency representations
- Cepstrum analysis (cepstral smoothed spectrum)
- Linear Prediction Coding (LPC) analysis
- Extraction of speech formant frequency candidates
- Frequency Band Analysis (band level)
- Fundamental Frequency Analysis

### 5.1.1 Pre-configured Viewer2 Settings

By default up to 12 individually designed STx Viewer2 configurations and analysis parameter settings can be implemented by the user in order to meet the special requirements of different applications. The configuration includes the layout of the viewer as well as colour palette settings, line style, font type and size, printer output configuration and
STx Spectrum Analysis

analysis default parameters. The pre-configured analyser methods offer commonly used spectrogram configurations:

The spectrogram viewer displays are usually laid out as follows:

1. The main spectrogram window (Graph 3), containing a quasi-3-dimensional plot of the squared frequency-amplitude over time.
2. The two sectioner zoom windows, containing a waveform magnification glass (Graph 1) and an amplitude spectrum display (Graph 2), both of which are centred on the currently active cursor position.
3. The segment waveform display (Graph 4), which shows the envelope of the segment under investigation.

Figure 27: Default spectrogram viewer display: Graph 1 – waveform zoom window Graph 2 – amplitude-spectrum display Graph 3 – main spectrogram window Graph 4 – segment waveform display Further graphs can be added, such as the parameter plot in Graph 5.

Other representations of the signal, such as fundamental frequency contour, formant frequency candidates or RMS values, can be displayed in additional traces (e.g. Graph 5, 6, etc.).
5.1.2 Analyse Startup Sequence

The most basic types of waveform display are activated via the View and Analyse functions. To display a soundsegment, you need to

- open a soundfile from the soundfile list or Findfile list window (either by a double click on the filename or, to open multiple files, by selecting them, right-clicking in the list pane and choosing Open from the pop-up menu)
- select a soundsegment from the list which appears when you open a soundfile
- choose Analyse from the Signal menu or from the right-click pop-up menu
- select the appropriate analysis method
- choose the appropriate analysis and display parameters
- save the parameter setup and start the program

Figure 28: The Analyse startup sequence: [1] select a soundfile from soundfile list window, [2] select a soundsegment from soundsegment list, [3] select Analyse from the Signal menu and [4] choose the appropriate analysis method. Note: if the method configuration and the parameter setup have been selected appropriately, steps [3 to 4] can be replaced by just activating View in step [3]. The program will immediately start the analysis using the current default parameter.

Shortcut: If the method configuration and the parameter setup have already been appropriately set, steps of Figure 29 (next page) can be omitted. Just select View, as in step [3] in Figure 28. The viewer is forced to apply the current analysis parameter setup to tagged soundsegments.
STx Spectrum Analysis

Figure 29: creation of a narrowband spectrogram, including analysis parameter setup control. The parameter setup dialog is obtained by double clicking on the selected method; each method has its own independent parameter setup. Currently the default spectrogram bandwidth is set according to FFT-frequency bins (and window length) to 22 Hz, %-overlap = 75 (frame shift) of FFT window. Frequency range = 0 to 10000 Hz, dynamic range = 50 dB up from -70 dB FS.

Once a spectrogram (or waveform) viewer has been configured to meet your requirements and the parameter setup has been saved, you can skip the Setup dialog and simply select View. The viewer will then be started using the pre-configured settings.

**Beware:** Multiple Viewer2 spectrograms can be displayed in a single STx session, either from the same or from different soundfiles. Nevertheless, for practical reasons (e.g. limited screen size, use of windows resources) it is assumed that no more than 8 viewers will be kept open simultaneously.

5.1.3 Analysis Parameter Settings

Almost all of the frequency-domain signal analysis and signal processing functions are configurable. FFT-frequency bin resolution (ΔHz) depends on the sampling frequency (Fs), FFT-frame and time window length. The absolute values of the Short-Time Fourier Transform (STFT) are usually squared and displayed as a function of time. The STx spectrogram can be configured with any possible combination of settings for (see sampling frequencies, frequency resolution and FFT-length on table 9.4 Annex):
• frequency scale
• frame length
• window length
• overlapping
• dynamic range
• smoothing

You can also adjust the display size and layout to meet your requirements. The STx graphics module automatically performs frequency and time axis interpolation on the basis of the input parameters. Viewer2 provides synchronized time alignment for all traces, even though different time frequency representations, spectrogram displays, Root Mean Square (RMS) amplitude, fundamental frequency contour, formant frequency candidates etc. have considerably different time windows, frame lengths and time shifts. All spectrogram and parameter displays of the type “abscissa = time” have automatic linear time standardisation to the window size specified in the Viewer2 setup, with the exception of the time zoom windows and where explicitly stated.

Not all the possible parameter settings are advisable for the optimal analysis and graphical representation of speech and music signals. Nevertheless, special parameter settings have not been excluded because experienced users can benefit from non-standard parameter settings especially for the analysis of non-speech or non-music or even non-acoustic signals. There are a number of default configurations, which would be sufficient for the new user’s needs. Implementing major changes to the viewer’s setup is supposed to be done step-by-step since there is a risk of unexpected results and even program termination if inconsistent setup values have been chosen. The experienced users may wish to use STx macros as templates for creating new menus, menu items and tailoring their STx systems for optimal results. Examples are given later on, to help the user chose their parameter settings for standard applications (see also STx setup section).

If you specify a desired frequency resolution, STx automatically adjusts the FFT signal frame parameters and the window length (i.e. time) accordingly. In order to assure computational efficiency, the FFT transformation lengths have been pre-defined as $2^n$. However, window lengths and the frame shift can be specified by the user. If the selected window length does not correspond to a power of two, it will be padded to the next highest power of two. Windows are used to convert signals of infinite duration into finite lengths; this is implemented through multiplication in the time domain (which is equivalent to a convolution in the frequency domain). A conversion to smoothly “switching” time windows is applied to the input signal prior to performing the FFT. This reduces leakage, an effect, which occurs when a rectangular window is used. Appropriate windowing suppresses the abrupt rise and fall of the waveform at the edges of a rectangular window.

Figure 30: Overlap-add and analysis frame (time) shift. Shifting the (time) window by half of its length enables loss-free reconstruction of the original waveform. Time shifts of > half of the window length will cause loss of information (and so should be used with caution).
and enhances the frequency analysis by avoiding the spread of spectral points into adjacent frequencies. There are many types of window, which serve various purposes and exhibit various properties (see TFR). When using FFT’s for spectral analysis, one should always bear in mind that a poor choice of window function can produce results that are quite different from what is expected.

5.2 FFT Narrowband Spectrograms (Music)

In order to obtain high-frequency resolution, large FFT frames and window lengths have to be used. However, the basic criterion for correct frequency reading still has to be met: namely that the signal be locally constant for the effective duration of the analysis window. Faster time varying signals exhibit smearing effects which lead to a deterioration in frequency and time resolution. The effective window length depends on the window’s shape; for most window types it can be approximated by calculating window duration/2.

Narrowband spectrograms are used for music signals, which are typically longer than speech signals (e.g. 7s or more). Most music analysis applications include melody contours, rhythm measurements and timbre/instrumentation analysis. For this purpose, the graphic and frequency resolutions must be matched. In order to resolve the individual partials of complex tones, the length of the FFT (or of any other frequency estimation method) has to be chosen correctly. Using an analysis frame length of 46 ms, most music signals can be analysed with a sufficient frequency resolution, i.e. without the appearance of substantial time smearing effects. Once again, the overlap factor (in %) controls the shift of the windowed analysis frame gliding over the signal segment, and determines the graphic time resolution of the spectrogram display. In the music analysis example given in Figure 31 below, this value is set to 50% in order to guarantee continuous overlap/add conditions.
5.3 FFT Wideband Spectrograms (Speech)

Wideband Spectrograms are typically taken from fast time varying signals such as speech utterances of short duration (typically 2s or less). At sufficiently high time resolution settings, the individual openings and closings of the glottis can be made visible. The appearance of vertical bars with spectral dominance in the formant frequency regions represents the periodicity of the voice fundamental frequency as well as the resonance behaviour of the vocal tract. In order to resolve individual periods of the fundamental frequency, the length of the FFT (or of any other frequency estimation method) has to be selected from within the range of the voice period duration. Fundamental frequency periods of male voices can usually be resolved using an analysis frame length of 11 ms or shorter. The overlap factor (in %, samples, ms) controls the shift of the windowed analysis frame gliding over the signal segment and determines the graphic time resolution of the spectrogram display. In the example shown in Figure 32 below, this value is set to a high level (90% overlap) to ensure the best graphical resolution on the time axis.
Figure 32: Widedband spectrogram of the utterance /enthusiasm/ [EnTj’u:ziaz7m], male speaker: bandwidth according to FFT-frequency bins (and window length) is set to 80 Hz, %-overlap = 90 (frame shift) of FFT window. Frequency range = 0 to 8000 Hz, dynamic range = 50 dB up from -70 dB FS. The appearance of vertical bars with spectral dominance in the formant frequency regions represents the periodicity of the voice fundamental frequency as well as the resonance behaviour of the vocal tract during voiced speech sounds. The left sectioner window shows the waveform centred at the active cursor position on the spectrogram, the right sectioner window shows the corresponding FFT magnitude as well as the LPC-smoothed spectrum. The parameter window (middle right) shows the time and frequency cursor values of the sectioner.

5.4 Cepstral Analysis

The short-term FFT spectrum serves as an intermediate step for producing more comprehensive representations of acoustic signals. There are many possible schemes for arranging a set of short-time spectra and spectral parameters. Furthermore, there are many natural sounds, which can be considered to be a product of two or more component signals or as a result of a combination of signals through convolution. For this reason it is often of interest to isolate the contributing signal components. This can be done if sufficiently detailed information about the individual components is given. One of frequently used mathematical tools for separating two convolved signals is provided by the cepstral analysis.
As an example, the source filter model of speech production can be seen as a decomposition of the speech signal, \( s_n \), into a (glottal) excitation \( e_n \) and a linear transfer function \( H(e^{\text{in}}) \), the resonance behaviour of the vocal tract. This can be modelled by convolution, expressed in the frequency domain:

\[
S(e^{\text{in}}) = H(e^{\text{in}})E(e^{\text{in}})
\]

By transforming this relation into the log domain, including z-transformation, the expression above can be written for speech as:

\[
\log(|S(e^{\text{in}})|) = \log(|H(e^{\text{in}})|) + \log(|E(e^{\text{in}})|)
\]

The method described was originally applied with great success as homomorphic deconvolution to seismic signals.

Figure 33: Functional diagram of cepstral analysis.

The basic operations of cepstral analysis (as implemented in STx) are:

1. capture acoustic signal frame
2. calculate appropriate data window for FFT
3. perform FFT
4. take log magnitude spectrum
5. perform the Fourier Transform (formally an inverse transform) on log magnitude spectrum
6. apply appropriate window on cepstrum (Lifter)
7. perform FFT on windowed cepstrum
The “spectrum” of the log amplitude spectrum of a waveform is called a CEPSTRUM (i.e. “spectrum” with the first 4 letters reversed). The abscissa of a cepstrum is called the QUEFRENCY. A filter applied in the quefrency domain (a window on the cepstrum) is called LIFTER and a harmonic component of the quefrency magnitude is called RAMONIC. Performing the FFT on the low quefrency liftered cepstrum (a low pass filter) results in the so-called SMOOTHED SPECTRUM, which can be seen as a maximum likelihood estimation of the original log amplitude spectrum of the signal under investigation.

Forward transform and backward transform without any modification in the cepstrum domain results in the original log magnitude spectrum. The degree of smoothing is determined by the window applied and the order of the cepstrum selected. None of the processing steps of the cepstral analysis needs any additional assumptions other than those necessary for the application of the FFT.

5.5 Linear Prediction Analysis

Linear Predictive Coding (LPC) was originally applied to the digital processing of speech signals, but under fulfilment of the necessary conditions it can also be used for spectral envelope estimation. The basic idea of LPC analysis is to estimate each sample of a waveform from a linear combination of \( p \) preceding values. The value of \( p \) is called the ORDER of the LPC.

According to the Acoustic Theory of Speech Production, the radiated speech signal can be seen as composed of the source spectrum (which is generated by the activity of the vocal folds) and the resonator function of the vocal tract cavities. Formant frequencies of speech depend in general on the shape of the vocal tract cavities, which is defined by factors such as the point of maximum constriction, which is controlled by the movement of the tongue, the degree of opening of the jaw and the position of the lips.
Linear Prediction Analysis

Figure 35: Log amplitude spectrum of a voiced speech segment, male voice. LPC smoothed spectrum superimposed. Number of LP coefficients (p) equals the sampling frequency of the signal in kHz plus 4 ("0"). The peaks of the LPC smoothed spectrum are used as formant frequency candidates.

The LPC method implements an acoustic tube model of the vocal tract. Cross-section area functions and reflection coefficients are obtained for the cylindrical sections of the acoustic tube model. It is worth mentioning that the standard LPC method assumes an all pole model of the vocal tract (no spectral zeroes), which is only justified for non-nasal voiced speech sounds. The implementation of a more elaborated model, covering poles and zeroes of the vocal tract including nasal cavities, is under development.
5.6 Formant Frequency Contour in Speech

The peaks of individual (voiced) frames of local LPC spectra are traced throughout the speech signal and are displayed in combination with the standard broadband spectrogram. This provides a convenient (and necessary!) graphical check on the reliability of the formant frequency extraction process. Due to the model restrictions mentioned above and various side effects (from signal quality and recording conditions) and due to the dependence of the formant extraction algorithm on the height of the fundamental frequency, visual inspection of the formant values obtained is highly recommended. The formant frequency candidate values (frequency and amplitude) are automatically written to the STx DataSet tables for further editing and processing.

5.7 Pseudo Wigner Distribution

Traditional time frequency representations (TFR) based on FFT algorithms suffer from reduced time resolution at high frequency selectivity. One way to overcome the problem of time-frequency dependence is to use alternative frequency estimation methods.
Figure 37: Viewer2-Pseudo Wigner Distribution spectrograms, comparable in frequency resolution to broadband FFT-analysis. Default Setup see below.

Figure 38.: Viewer2 Pseudo Wigner Distribution spectrograms. Default Setup: frame length: 25ms overlap: 99% smoothing: 3ms FIR-filter length: 16ms.
5.8 Frequency Band Analysis

Short-term FFT amplitude spectra are the basis for spectrograms of time varying signals, as described above. They can be viewed as single-amplitude spectra representing the frequency content of a windowed signal at a given point in time (also described above); or, alternatively, frequency bins can be displayed (as a function of that frequency over time). Roughly speaking, the FFT can be regarded as a fine-grained filter bank, with each of its frequency bins representing a narrowband filter, which delivers the magnitude (level) in each frequency channel (band) over time.

![Figure 39: Frequency Band Signal Energy: the middle trace shows the 10-band signal energy as a function of time. The frequency bands can be defined in the Setup dialog window (shown in the upper right corner).](image)

Due to the characteristics of the FFT, the frequency resolution is linear and depends on the window and transformation length selected. But many applications require logarithmic frequency spacing or even arbitrary frequency band analysis. This is why STx provides the option of specifying frequency bands of freely selectable width, so as to deliver the composed magnitude of all FFT bins that fall within the defined frequency range(s). By selective definition of frequency bands discriminative spectral sieves can be implemented for acoustic feature extraction (e.g. vowel sieves). Preferred frequencies and frequency bands are shown in Error! Reference source not found. Error! Reference source not
The frequency band table is divided into whole (1/1), half (1/2) and third (1/3) octaves (see Preferred Frequencies and Frequency Bands table 9.9).

5.9 Short-Time Spectrum and Spectrum Averaging

Short time spectrum analysis and spectrum averaging is performed by applying the options of the Spectrum menus of the Analyse section, straightforward; the procedure is quite similar to the handling of the amplitude spectrum in the Spectrogram Viewer2's sectioner window. All precautions for the use of FFT-analysis, which have been described for creating spectrograms, equally apply for this section. Especially spectrum averaging has to meet the necessary condition the signal being locally steady state for the duration of the averaging period. Otherwise the resulting averaged amplitude spectrum does not represent the spectral content of the signal properly. Averaging a frequency sweep of a sinusoid for instance would produce a broad band spectrum which looks like steady state broad band noise. Nevertheless if the steady state condition of a signal is adequately met, averaging provides the advantage to suppress random noise and to enhance periodic signal components substantionally. That is the most frequent reason for applying spectrum averaging on noisy signals.

Averaging is provided for almost all frequency analysis methods available in STx. The Spectrum section of the Analyse menu offers the user currently the selection from FFT magnitude (default), Cepstrum and LPC smoothed spectrum as well as combinations of both, FFT+Cepstrum or FFT+LPC; from the averaging options peak, linear and exponential averaging can be specified.

Figure 40: Linear averaging of the quasi steady state part of the phoneme /ea/. The LPC smoothed spectrum is overlaid on the FFT amplitude spectrum.
As most parameters computed in STx single spectra or averaged spectra are stored automatically in the STx DataSet. For further processing and use they can be edited by means of the DataSet Editor. The Spectrum Editor provides functions such as spectrum invert, graphic editing of the spectrum, copy, paste, interpolation etc.

The amplitude spectrum of a noise background is inverted to generate the frequency response of a (multi-band) spectrum filter. Filtering a signal with its inverted amplitude spectrum is used to cancel unwanted signal parts (see Signal Processing, Spectral Subtraction).
6. Signal Processing of STx Sound-Segments

In contrast with the rest of STx, the functions described in this chapter can be (and usually are) destructive.

*Beware: The Signal Processing functions of STx are intentionally designed to alter the signal content of soundfiles*

The signal processing functions of STx are called from the Processing menu of the STx SOUNDFILE LIST window. Use them with care! We highly recommend that you copy the whole soundfile or selected soundsegments before any signal processing function is started! STx by default does not warn the user if the source and destination signal addresses of a soundsegment are specified as the same and therefore overwriting of an existing signal is performed. Currently no (!) undo function is available! Deleted files are not placed in the Windows recycle bin, but are physically overwritten on the harddisk!

6.1 Signal Copy

The generally applicable Signal Copy function of STx is called from the Processing Menu Section of the STx SOUNDFILEDIRECTORY window. It is worth mentioning that the STx copy function as described below is inherently applied in all Signal Processing and Signal Modification sections of STx involving signal source and destination operations. Copy provides copy and paste as well as several signal overwriting options and therefore requests the exact specification of source and destination soundfile addresses. The Process Segment(s) dialog box provides the following menu items:
Signal Processing of STx Sound-Segments

Figure 42: The Process Segment(s) dialogue box provides the main entry for setup and start of STx Signal Processing functions currently available. Note: these functions are intentionally changing the content of soundfile(s).

- Define a New target soundfile: create a new soundfile and copy the selected soundsegment(s) into this file
- Specify the name of the source file and the soundsegment(s) (source signal), which usually is selected from the SoundfileDirectory window or alternatively by explicit input in the dialog field(s)
- Specify the target soundfile which can be selected from the options:
  - source file (copy into the same file)
  - compatible existing files (copy into soundfiles compatible in format, sampling rate and binary wordlength)
  - all (other) files
- Define the target address(es), selected from the options:
  - replace the source signal (Caution: this is definitively overwriting of the source soundsegment!)
  - append (the copy) to the target file (which actually can be the currently opened source file or any other soundfile)
  - specify any target address in the target soundfile (Caution: this can be total or partial overwriting of already existing soundsegments!)
- Specify the target soundsegment attributes, Name and Extension(s)
Reverse signal provides the option to copy soundsegments time inverted in order to play and process them from back to forward.

### 6.2 Amplify / Limit / Normalize

The Amplify/Limit function of STx enables the user to control the amplitude of a time domain signal after it has been recorded in a soundfile. The gain factor can be specified either linear as a rational number or in +/- dB. If appropriate, the waveform can be differentiated and/or inverted on the magnitude scale in the same processing step.

The Limit function setup requests the specification of the maximum waveform value and the onset point (absolute, relative or in dB) of the limiter function. The limiter function itself can be selected from the options: rectangle, arctan and exponential.

Normalize is used in order to control the waveform magnitude of segments located in the same or in different soundfiles. Normalization on Peak or RMS values can be selected.

### 6.3 Automatic Gain Control (AGC)

Automatic Gain Control is a nonlinear process, usually applied to waveforms containing two or more signals, such as a telephone conversation, and one signal source is considerably weaker than the other.

By proper selecting the parameters of the AGC function a significant increase of intelligibility can be obtained. As each of the applications requests its own parameter settings due to different recording conditions, no general rule can be given; nevertheless the default settings, as displayed above, have proven as a reasonable starting point.

![AGC Settings](image)

*Figure 43: Default settings of the AGC function: The user is invited to start with these default settings. As no generally applicable rules can be given, optimization of the AGC processing has to remain empirical.*
6.4 Digital Filtering: LP-HP-BP, Spectrum

Today digital filters provide convenient tools for almost all signal manipulations in the frequency domain. As this manual does not include a description of digital filter theory, the reader is invited to consult tutorials on this topic freely available at many Internet addresses. However, STx digital filter section introduces into the primitives of main and side effects appearing from the practical view of application. From experience, the user is advised to apply filtering on signal copies only and to control the filtering carefully by means of STx spectral analysis tools such as Viewer2 spectrogram and sectioner as well as spectral averaging. Before (if ever) original signals are processed, the user is encouraged to examine the test results by listening. Auditory control still provides a fast and convenient way for the detection of filter mistuning or distortion.

Digital filters simulate the transducer behaviour of analogue filters to a large extend. They introduce relatively small insertion loss to waves in one or more frequency bands and relatively large insertion loss to waves of other frequencies. STx filter types can be selected from the following methods: Wiener filter no overlapping, Wiener filter with half overlapping and Phase Vocoder. For all of the filter types the length of the filter, which determines the frequency bin resolution, can be specified in samples or in ms.

6.4.1 Low-pass Filter

The low-pass filter has a single transmission band which ranges from \( f_1 \) near to 0 Hz to some upper band-edge frequency, \( f_2 < \) sampling frequency/2.

6.4.2 High-pass Filter

The high-pass filter has a single transmission band which ranges from \( f_1 \), some band-edge frequency greater than zero up to the half of the sampling frequency, \( f_2 \).

6.4.3 Bandpass Filter

The bandpass filter has a single transmission band which ranges from a lower band-edge frequency \( f_1 > 0 \) to an upper band-edge frequency \( f_2 < \) sampling frequency/2.

6.4.4 Spectrum- (Multi-band-) Filter

The spectrum- or multi-band-filter has a theoretical unlimited number of transmission bands in the frequency range from 0 Hz to sampling frequency/2. In praxis, the frequency response of a multi-band-filter is limited by the length of the filter and the frequency bin resolution. STx provides the convenience to use a previously calculated amplitude spectrum as a multiband-filter. Graphic editing of the spectrum allows the creation of arbitrary filter frequency responses.
Note: due to the limited length of digital filters special care has to be taken when steep transition bands are specified. They result in unreal filter responses producing a number of side effects.

6.5 Denoising (Spectral Subtraction)

Noise reduction systems frequently use spectral averaging and adaptive filters (spectral subtraction). It is advisable to prewhiten the broadband background noise by adding its inverted spectral magnitude. The frequency spectrum of the background noise is obtained by averaging short time spectra of silent segments of a recording. If system transfer functions, such as horn resonances of historical sound recording devices are known, corresponding correction filter(s) can be included in this processing step. The STx denoising module takes advantage from the difference between the statistical characteristics of the noise and the signal. Noise, such as surface noise of a historical sound recording, is added to the signal and has frame to frame randomness. The signal is assumed to remain locally stable to that extent, that its amplitude spectra resemble frame to frame. The gliding spectral average is used to generate an adaptive filter, which is applied to the prewhitened input signal. The degree of noise reduction depends on the count of averaging steps and the length of the short time frames. A balance between the range of averaging and the nonstationarity of the signal has to be found in order to avoid time smearing effects.

Figure 44: functional diagram of the STx denoising module using spectral averaging and spectral subtraction.
6.6 Irrelevance Filter "What You See is What You Hear!"

Research on hearing theories has developed consistent models of simultaneous masking. Masking is the psychoacoustical process by which the threshold of audibility of a sound is raised by the presence of another (masking) sound. The masking customarily is expressed in decibels. STx provides a computational model for evaluating the masked threshold of any running audio signal. The signal is split up into two spectral layers: the lower one containing the masked spectral components and the upper one, which holds all unmasked spectral components. By subtracting the masked spectral components from the original sound, spectrograms can be created which show the auditory relevant information only. After filtering all masked signal parts the remainder contains no psycho-acoustic irrelevant signals anymore, it therefore can be assumed as close to the excitation signal delivered by the auditory periphery to higher auditory centers.

Using the Phase Vocoder method for filtering offers the additional option of time compression and expansion if specified. Time expansion can be useful in zooming in into fast transients, which can not be perceived analytically at normal speed. A time expansion factor of 2 has been experienced as appropriate for many music and speech applications.

6.7 Overmasking (Spectral Suppression)

This part of the documentation is currently in preparation. Please contact us for further details.
7. The STx Sequencer

Individual soundsegments can be chained together to form a segment SEQUENCE. A segment sequence can contain any segments (whether pre-defined or created on the fly) from any accessible soundfiles. Once defined, a sequence is treated by STx as if it were a single soundsegment: it can be played, analysed and otherwise manipulated in the same way as a single soundsegment.

7.1 Creating a Segment Sequence

To use the sequencer, first open the segment list window and select the segment which you want to be the first segment of the sequence. Then select Segment – Sequence (or Sequence from the right-button pop-up menu). (NB: If no segment is selected, the sequencer has to be opened from the Tools pull down menu of the soundfile list window). This opens the Select a Sequence window, as shown in Figure 45.

![Select a Sequence window](image)

*Figure 45: The Select a Sequence window when the first segment has been selected to create one’s first sequence.*

Click on the New button, and a new sequence is created with the selected segment as its first item, as shown in Figure 46. Until you save the sequence in a setup file (*.stu), the sequence has the default name "#1".
The STx Sequencer

Figure 46: The first segment (here: My1stSegment, from the soundfile F:\work\RB2.wav) of the first sequence created.

You can now add a further segment to the sequence by selecting it in the segment list window (of the same soundfile, or of a different soundfile) then clicking on Segment – Sequence (or Sequence from the right-button pop-up menu). Now the Select a Sequence window offers you the choice of selecting an existing sequence or creating a new one, as shown in Figure 47.

Figure 47: The Select a Sequence window when a first sequence already exists. Click on Select to add the soundsegment to sequence "#1" or New to create a new sequence.

In Figure 48 you see an example of a first sequence which contains 4 segments from 2 different soundfiles.

Figure 48: A sequence containing multiple soundsegments from two different soundfiles.
7.2 Managing and Using Segment Sequences

Once a sequence has been created, it can be saved or loaded (via the File menu). The individual segments can be re-ordered (using Copy, Cut, Paste from the Segments menu) or edited (using Edit from the Segments menu). Beware: when re-ordering segments, Paste overwrites an existing segment entry; it is useful to add empty segments to the sequence (using Special – Set Size and increasing the number of segments as desired).

Before you can Play, View or Analyse a sequence (from the Signal menu of the sequence window) you must ensure that empty segment entries are removed, and that the sequence is internally collated by STx (using Signal – Test).

7.3 Creating a Sequence of Synthesized Sounds

Calling the Sequencer from the Tools pull down menu of the soundfile list window enables the user to create sequences of synthesized sounds, such as white noise and filtered noise, or several other “primitive” synthesis products available from the sequencer Edit Signal window (double click on a sequence list line) combined with segments from a soundfile. The following screenshot displays the concatenation of examples of filtered noise and a speech segment to a sequence, which can be accessed as a continuous signal. This sequence can furthermore be copied in an existing soundfile.

Figure 49: This sequence is composed of synthesized signals as well as of segments taken from a soundfile.
Sequences can be composed of sound segments stored in disk files as well as combined and mixed with sounds synthesised in real time. Experienced users may take advantage from this feature, especially designed to be used in adaptive psychoacoustic test procedures. Test and comparison sounds can be adapted according to the subject’s response.

Various advanced functions are available from the Segments – Settings and from the Special menu. These functions are under constant development, and we encourage the advanced user to try them out. They will be more fully documented when their full extent and structure has been decided upon.
8. STx DataSet

8.1 DataSet Handling and Export of Parameters

The STx DataSet section provides an elaborated interface for storing, editing and exporting parameters and raw data computed by means of automated STx signal-processing functions as well as obtained from signal measurements and in build basic statistics. The DataSet’s graphic-acoustic editor supports manual and computational post processing of all data in order to enable outlier rejection and corrections of errors such as octave jumps (of fundamental frequency values), discontinuities in parameter contours, wrong assignments to formant traces, editing of amplitude spectra etc. As an important feature of STx DataSets the stored raw data remain fully referenced to the original soundfiles and signal segments from which they have been computed even after being edited. This enables the user to maintain interactive auditory, graphic and numeric control during the whole working period of a project and to apply additional analysis steps if appropriate.

Figure 51: Opening of the DataSet from the Tools pop down menu.

The DataSet dialogue is opened from the Tools pull down menu of the STx soundfile list window. From the beginning of a new software installation the DataSet window is empty as long as no STx function has been performed which by default delivered raw data in the current or in previous STx sessions.
8.2 Working with DataSets

Assume a Wideband Spectrogram has been taken from one or more speech sound segments of appropriate duration (typically 2s or less) using one of the spectrogram functions provided in the Analysis section of STx Viewer2. Formant frequency candidates have been extracted by picking the peaks of Linear Prediction Coding (LPC) amplitude spectra (vocal tract transfer functions computed by means of the autocorrelation method). These raw data of formant traces are superimposed on the wide band spectrogram display in order to provide control on their validity. Furthermore an estimate of the fundamental frequency contour has been obtained by means of SIFT (Simplified Inverse Filtered Technique). The estimate of the fundamental frequency values appear in a separate window below the spectrogram display just above the envelope of the waveform.

![Figure 52: Speech Spectrogram with formant frequency candidates overlaid and fundamental frequency contour containing downward octave jumps due to high pitched male voice; voice setting close to singing, music instrument tone background (guitar).](image)
8.2.1 Speech Parameters are stored in the DataSet

All together, the estimates of the fundamental frequency (F0A), the formant frequency candidates (FO1-FO5) as well as further parameters optionally computed from the signals under consideration are stored automatically in a DataSet File (default.sd1) of the current STx session.

![Figure 53: List of DataSet entries of the current STx session.](image)

As automatically computed raw data of fundamental frequency and formant extraction algorithms occasionally are subject to post processing a DataSet graphic Editor is provided. By selecting the appropriate parameter set the Edit functions in the popup window box shown (right mouse button) are available. If activated they are applicable on the parameter values bracketed by the rectangle between both time line cursors (cross-hairs) on the parameter window(s).

![Figure 54: View starts the display and Edit section of the DataSet Editor after selection of the parameter sets to be edited (toggle ON/OFF).](image)

From practice it is well known that the fundamental frequency extraction algorithm works accurate on clean speech signals but fails occasionally at high pitched voices or weak source signals. Our example demonstrates a further complication by a voice setting close to singing and the presence of interfering background sounds. Nevertheless the occurring errors are limited to a few easily detectable octave jumps and some voiced frames which should be unvoiced. The errors are corrected stepwise by means of the DataSet editor functions “multiply by factor 2” and “erase values in cursor box”.

![S_TOOLS-STx USER’S GUIDE](image)
Figure 55: Corrections of wrong values of fundamental frequency raw data by means of the interactive DataSet editor. Values in double line frame boxes are corrected by multiplying with the factor 2. Single line frame box values are set to zero.

Figure 56: The corrected and cleaned fundamental frequency contour is ready for statistical post processing.
8.2.2 Export of Parameter Values after Editing (DataSet)

STx makes already available a limited basis set of statistics and parameter displays. Nevertheless higher level statistics and further post processing is frequently necessary. In order to support a general parameter interface for external programs the DataSet section of STx has been equipped with convenient export options.

The Tools Section of the DataSet window provides the Export function for parameters computed and edited previously and finally stored in the DataSet. Automatic output file naming is provided. Values are written in various formats including the signal time scale or a running index; delimiter options as well as missing values can be specified. Parameter tables are optionally merged into a single sheet. Frequently ASCII file format is selected.

Select Parameters enables the specification of the parameter set(s) actually to be exported from the current data (right mouse button popup menu). The data are stored on the destination file as specified (c:\OEAW\STX\0002.asc).

For parameter sets prepared to be used in Microsoft Excel, Mathcad and Axum support for further processing in statistics and graphics display as well as on input in a database can be provided on request.
9. Appendix

9.1 Defining Soundsegments

A soundsegment is a contiguous part of sound data in a soundfile. It is defined by SEGMENTNAME, SEGMENTSTARTADDRESS and SEGMENTDURATION. The SegmentStartAddress is relative to the beginning of the soundfile, and the SegmentDuration begins at the SegmentStartAddress.

![Figure 58: Input window for a new entry in the segment directory: SegmentName (Name) SegmentStartAddress (Begin/Task) SegmentDuration (Length) and optional annotations (here: Typ, Text and Tag).]

STx also supports various absolute and relative time specifications (detailed in the next subsection) for the fine grained definition and location of an audio segment in an opened soundfile. In this way, any segment address using a well-formed address expression can be entered in the Begin/Task field of the Create New Segment window. (In the Create New Segment window STx automatically converts a well-formed address to the StartAddress – Duration format.)

**Note:** In order to enable the use of +/- in arithmetic operations, the underline (_) character, rather than a hyphen (-), is employed to specify the “from-to” relation in the specification of segment address expressions. STx accepts a hyphen character in place of a true minus character (on the numerical keyboard).
9.2 Specifying Soundsegments on the Fly

Soundsegments do not need to be pre-defined using the Create New Segment menu. They can be specified on the fly using addresses which refer to either a) positions within the soundfile or b) positions relative to some already defined soundsegment.

Some examples of valid segment addresses are given in Table 6 (absolute addresses) and Table 7 (relative addresses and expressions).

Table 6: Valid absolute segment address (time) specifications (to be entered in the Begin/Task field of the Create New Segment window)

<table>
<thead>
<tr>
<th>from</th>
<th>to</th>
<th>sample expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>sample#</td>
<td>sample#</td>
<td>48000_192000</td>
</tr>
<tr>
<td>ms</td>
<td>ms</td>
<td>1000ms_4000ms</td>
</tr>
<tr>
<td>s</td>
<td>s</td>
<td>1s_4s</td>
</tr>
<tr>
<td>beginning of segment</td>
<td>end of segment</td>
<td>SegmentName</td>
</tr>
</tbody>
</table>

Table 7: Valid relative segment addresses and address expressions (to be entered in the Begin/Task field of the Create New Segment window)

<table>
<thead>
<tr>
<th>from</th>
<th>to</th>
<th>sample expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>sample#–#</td>
<td>sample#++</td>
<td>48000–22050_192000+48000</td>
</tr>
<tr>
<td>ms–ms</td>
<td>ms+s</td>
<td>1000ms–500ms_4000ms+1s</td>
</tr>
<tr>
<td>s–sample#</td>
<td>s+ms</td>
<td>1s–22050_4s+1000ms</td>
</tr>
</tbody>
</table>

In addition, the time pointers and identifiers in Table 8 can be used in address specifications or expressions.

Table 8: Time pointers and identifiers for segment addresses (to be entered in the Begin/Task field of the Create New Segment window)

| SegmentName:B = | first sample of SegmentName |
| SegmentName:E = | last sample of SegmentName |
| SegmentName:L = | duration (length) of SegmentName |

Note: Any type of absolute or relative address or pointer/identifier can be combined with any other address type in the specification of a soundsegment.

9.2.1 Soundsegment Addressing throughout All STx Modules

The syntax used in the definition of soundsegments is also used in other modules of the STx system. Thus direct soundsegment addressing can be used with the Play..., View..., Analyse... and other commands.

9.2.1.1 Example

Open any (copy of a) wave file that is longer than 14 minutes in length, and with Segments – New define My1stSegment as 120s to 240s (i.e. 120s_240s – note how
STx automatically converts this to the Name – Duration format. Create a second soundsegment with the name My2ndSegment from 600s to 840s. Then select play... from the Signal menu, and in the Segment field of the Play – Enter Segment window, enter the “segment address” given in the following table in order to hear the stretch of signal in the “what happens” column.

<table>
<thead>
<tr>
<th>segment address</th>
<th>what happens</th>
</tr>
</thead>
<tbody>
<tr>
<td>My1stSegment</td>
<td>plays 120s to 240s</td>
</tr>
<tr>
<td>My2ndSegment</td>
<td>plays 600s to 840s</td>
</tr>
<tr>
<td>My1stSegment:B_My1stSegment:E</td>
<td>plays 120s to 240s</td>
</tr>
<tr>
<td>My1stSegment:B_My2ndSegment:E</td>
<td>plays 120s to 840s</td>
</tr>
<tr>
<td>My1stSegment:B-60s_My2ndSegment:E+2s</td>
<td>plays 60s to 842s</td>
</tr>
</tbody>
</table>

Similarly, the commands view..., analyse... etc.), associated with the corresponding address specifications, will be executed on the same soundsegments.

### 9.3 Soundsegment Directory File Template(s)

Each soundsegment has a unique name within a soundsegment directory file. You can define soundsegments in a number of different ways, and can customise your segment template to incorporate precisely the information you want to associate with your soundsegments. Examples of additional information usually stored are lexical transliteration, phonetic transcription as well as technical data associated with soundsegments. Customisation of the segment template is carried out via the Setup menu in the soundfile list window. From this menu, select Segment Template, and the Segment Template Management window opens. Here you can modify the contents of the default template, then save it under a different name. For an explanation of the options available for each entry in the template see figures below.

![Figure 59: open the soundsegment directory file template management.](image-url)
Appendix

Figure 60: soundsegment directory files can be configured to meet the requirements of virtually all database systems. The number of fields for annotation can be extended to any size necessary.

9.4 Frequency resolution & window lengths

Table 9: Frequency bin resolution (ΔHz) depending on sampling frequency (FS), FFT-frame and time window length.

<table>
<thead>
<tr>
<th>Fs (Hz)</th>
<th>FFT length (samples)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>16384</td>
</tr>
<tr>
<td>8000</td>
<td>0.48</td>
</tr>
<tr>
<td>1025</td>
<td>0.67</td>
</tr>
<tr>
<td>16000</td>
<td>0.97</td>
</tr>
<tr>
<td>22050</td>
<td>1.34</td>
</tr>
<tr>
<td>32000</td>
<td>1.95</td>
</tr>
<tr>
<td>44100</td>
<td>2.69</td>
</tr>
<tr>
<td>48000</td>
<td>2.93</td>
</tr>
<tr>
<td>64000</td>
<td>3.90</td>
</tr>
<tr>
<td>88200</td>
<td>5.38</td>
</tr>
<tr>
<td>96000</td>
<td>5.85</td>
</tr>
</tbody>
</table>
Table 10: Effective window duration (ms), depending on sampling frequency (FS), FFT-frame and time window length

<table>
<thead>
<tr>
<th>Fs (Hz)</th>
<th>FFT length (samples)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>16384</td>
</tr>
<tr>
<td>8000</td>
<td>1024.00</td>
</tr>
<tr>
<td>11025</td>
<td>743.03</td>
</tr>
<tr>
<td>16000</td>
<td>512.00</td>
</tr>
<tr>
<td>22050</td>
<td>371.51</td>
</tr>
<tr>
<td>32000</td>
<td>256.00</td>
</tr>
<tr>
<td>44100</td>
<td>185.76</td>
</tr>
<tr>
<td>48000</td>
<td>170.66</td>
</tr>
<tr>
<td>64000</td>
<td>128.00</td>
</tr>
<tr>
<td>88200</td>
<td>92.88</td>
</tr>
<tr>
<td>96000</td>
<td>85.33</td>
</tr>
</tbody>
</table>

**Note:** in order to obtain the frequency resolution values and window lengths listed above in an STx analysis, specify either the Δf-value bandwidth [Hz] or spectrogram frame length [ms]² in the Analyse – Setup dialogue of the soundsegment list window.

### 9.5 Long-duration Segment Analysis

STx processing speed is mainly determined by graphics setup parameters. Generally, high graphics resolution is necessary if signal durations of ≤ 10s are involved. If larger signal durations are processed, the following setup changes can be performed in order to speed up the analysis of long soundsegments without loss of display quality.

### 9.6 Waveform Display – Viewer 1

Viewer 1 provides the setup option timebar only. This option disables STx to draw a highly compressed waveform envelope of the signal contained in the whole soundfile currently opened, a process which can be time consuming. Enabling timebar only provides a ruler line instead of waveform display which supports the user by specifying sound segments on a timebar basis mainly.
### Analysis – Viewer 2

Spectrograms use different FFT time window lengths in order to provide appropriate frequency resolution. Basically, a 50% frame overlap of consecutive time windows is sufficient for proper frequency analysis. Nevertheless, broadband analysis, which usually is applied on signals of durations $\leq 3s$ with a bandwidth of typically 100 Hz, produces better graphics quality if the frame overlap parameter is raised to 90%. In case of long-duration signals ($\geq 10s$) and analysis bandwidth of typically 10 Hz, the overlap percentage parameter can be reduced to 60% without loss of graphics resolution. This improves the graphics display speed considerably.

![Figure 61: Waveform Display – Viewer 1. Checking timebar only speeds up the waveform display for signals of long duration.](image)
9.8 Audio Levels and Amplitude Measurements

9.8.1 Voltage and Audio Signal Levels

Table 11. Commonly used Voltage and Audio Signal Levels and their reference values. Note: 0 dBm = 1 milliwatt which is equivalent to 0.7746 volts RMS into a 600 Ω load.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Level Definition</th>
<th>0 dB = Reference Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>dBV</td>
<td>$20 \times \log(V/V_{ref})$</td>
<td>$0 \text{ dBV} = V_{ref} = 1 \text{ V}$</td>
</tr>
<tr>
<td>dBm</td>
<td>$10 \times \log(W/W_{ref})$</td>
<td>$0 \text{ dBm} = W_{ref} = 1 \text{ mW}$</td>
</tr>
<tr>
<td>dBu</td>
<td>$20 \times \log(U/U_{ref})$</td>
<td>$0 \text{ dBu} = U_{ref} = 0.7746 \text{ V}$</td>
</tr>
<tr>
<td>dB SPL</td>
<td>$20 \times \log(p/p_{ref})$</td>
<td>$0 \text{ dB SPL} = p_{ref} = 20 \mu\text{Pa}$</td>
</tr>
</tbody>
</table>

-10 dBV = $10^{10} \times 10^{20} \times 1 \text{ V} = 0.316 \text{ V}$
-7.8 dBu = $10^{7.820} \times 0.7746 \text{ V} = 0.316 \text{ V}$
0.7746 V across 600 Ω: $0.77462 / 600 \Omega = 1 \text{ mW}$
+4 dBu = $10^{4} \times 10^{1} = 0.7746 \text{ V} = 1.2276 \text{ V}$

Note: dBV = dBu (dBu preferred) all voltage measurements in $V_{rms}$
Appendix

Table 12. Full Scale Range (dB FS) available at different digital word lengths (bits/sample).

<table>
<thead>
<tr>
<th>Coding bits/sample: n</th>
<th>8</th>
<th>16</th>
<th>18</th>
<th>20</th>
<th>22</th>
<th>24</th>
<th>32</th>
</tr>
</thead>
<tbody>
<tr>
<td>FS ratio: N/1</td>
<td>$2^n$</td>
<td>$2^{16}$</td>
<td>$2^{18}$</td>
<td>$2^{20}$</td>
<td>$2^{22}$</td>
<td>$2^{24}$</td>
<td>$2^{22}$</td>
</tr>
<tr>
<td>FS ratio: dB</td>
<td>48</td>
<td>96</td>
<td>108</td>
<td>120</td>
<td>132</td>
<td>144</td>
<td>192</td>
</tr>
</tbody>
</table>

Note: dB (FS) = $20 \times \lg(N) = 20 \times \lg(2^n) = 20 \times 0.3010 \times n$

9.8.2 Amplitude Measurements, Root Mean Square Value, $V_{\text{RMS}}$

The RMS value of an alternating current is that current which will give the same heating effect as the equivalent direct current. The RMS value of $y=f(x)$ over the range $x=a$ to $x=b$ is given by:

$$\text{RMS value} = \sqrt{\frac{1}{b-a} \int_a^b y^2 \, dx}$$

The mean or average value of a waveform between $x=a$ to $x=b$ is given by:

$$\text{AVG value} = \frac{1}{b-a} \int_a^b y \, dx$$

For simple tones (sinusoidal waveforms) the relationships between RMS value, AVG value, peak and peak-to-peak value are given in Table 13.

Table 13: Peak-to-peak, RMS and AVG value conversion for sinusoidal waveforms:

<table>
<thead>
<tr>
<th>from value</th>
<th>multiplication factor to value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>average</td>
</tr>
<tr>
<td>average</td>
<td>1.0</td>
</tr>
<tr>
<td>RMS</td>
<td>0.9</td>
</tr>
<tr>
<td>peak</td>
<td>0.637</td>
</tr>
<tr>
<td>peak-to-peak</td>
<td>0.32</td>
</tr>
</tbody>
</table>
Audio Levels and Amplitude Measurements

Figure 63: Peak-to-peak, RMS and AVG values of sinusoidal waveforms.

Figure 64: Peak-to-Peak, RMS and AVG values of simple waveforms.

Figure 65: Peak-to-Peak, RMS and AVG values of special waveforms.
## 9.9 Preferred Frequencies and Frequency Bands

Table 14: Table of Preferred Frequencies and Frequency Bands (see section 5.8)

<table>
<thead>
<tr>
<th>Octave (Hz)</th>
<th>Octave (Hz)</th>
<th>Octave (Hz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>16 x x x</td>
<td>160 x</td>
<td>1600 x</td>
</tr>
<tr>
<td>18</td>
<td>180 x</td>
<td>1800</td>
</tr>
<tr>
<td>20 x</td>
<td>200 x</td>
<td>2000 x x x</td>
</tr>
<tr>
<td>22.4 x</td>
<td>224</td>
<td>2240</td>
</tr>
<tr>
<td>25 x</td>
<td>250 x x x</td>
<td>2500 x x x</td>
</tr>
<tr>
<td>28</td>
<td>280</td>
<td>2800 x</td>
</tr>
<tr>
<td>31.5 x x x</td>
<td>315 x</td>
<td>3150 x</td>
</tr>
<tr>
<td>35.5</td>
<td>355 x</td>
<td>3550</td>
</tr>
<tr>
<td>40 x</td>
<td>400 x</td>
<td>4000 x x x</td>
</tr>
<tr>
<td>45 x</td>
<td>450</td>
<td>4500</td>
</tr>
<tr>
<td>50 x</td>
<td>500 x x x</td>
<td>5000 x</td>
</tr>
<tr>
<td>56</td>
<td>560</td>
<td>5600 x</td>
</tr>
<tr>
<td>63 x x x</td>
<td>630 x</td>
<td>6300 x</td>
</tr>
<tr>
<td>71</td>
<td>710 x</td>
<td>7100</td>
</tr>
<tr>
<td>80 x</td>
<td>800 x</td>
<td>8000 x x x</td>
</tr>
<tr>
<td>90 x</td>
<td>900</td>
<td>9000</td>
</tr>
<tr>
<td>100 x</td>
<td>1000 x x x</td>
<td>10000 x</td>
</tr>
<tr>
<td>112</td>
<td>1120</td>
<td>11200 x</td>
</tr>
<tr>
<td>125 x x x</td>
<td>1250 x</td>
<td>12500 x</td>
</tr>
<tr>
<td>140</td>
<td>1400 x</td>
<td>14000</td>
</tr>
<tr>
<td>160 x</td>
<td>1600 x</td>
<td>16000 x x x</td>
</tr>
</tbody>
</table>
9.10 Frequencies of Musical Notes

Table 15: FREQUENCIES OF THE EQUALLY TEMPERED SCALE; BASED ON THE INTERNATIONAL STANDARD A = 440 HERTZ

<table>
<thead>
<tr>
<th>Note</th>
<th>S</th>
<th>f</th>
<th>2πf</th>
<th>Note</th>
<th>S</th>
<th>f</th>
<th>2πf</th>
<th>Note</th>
<th>S</th>
<th>f</th>
<th>2πf</th>
</tr>
</thead>
<tbody>
<tr>
<td>C0</td>
<td>0</td>
<td>16.352</td>
<td>102.74</td>
<td>C3</td>
<td>36</td>
<td>130.81</td>
<td>821.92</td>
<td>C6</td>
<td>72</td>
<td>1,046.6</td>
<td>6,575.4</td>
</tr>
<tr>
<td>1</td>
<td>17.324</td>
<td>102.74</td>
<td>37</td>
<td>138.59</td>
<td>870.79</td>
<td>D6</td>
<td>74</td>
<td>1,108.7</td>
<td>6,966.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D0</td>
<td>2</td>
<td>18.354</td>
<td>115.32</td>
<td>D3</td>
<td>38</td>
<td>146.83</td>
<td>922.58</td>
<td>D7</td>
<td>76</td>
<td>1,174.7</td>
<td>7,380.6</td>
</tr>
<tr>
<td>3</td>
<td>19.445</td>
<td>122.18</td>
<td>39</td>
<td>155.56</td>
<td>977.43</td>
<td>E6</td>
<td>75</td>
<td>1,244.5</td>
<td>7,819.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E0</td>
<td>4</td>
<td>20.602</td>
<td>129.44</td>
<td>E3</td>
<td>40</td>
<td>164.81</td>
<td>1,035.6</td>
<td>E6</td>
<td>76</td>
<td>1,318.5</td>
<td>8,284.4</td>
</tr>
<tr>
<td>F0</td>
<td>5</td>
<td>21.827</td>
<td>137.14</td>
<td>F3</td>
<td>41</td>
<td>174.61</td>
<td>1,097.1</td>
<td>F6</td>
<td>77</td>
<td>1,396.9</td>
<td>8,777.1</td>
</tr>
<tr>
<td>6</td>
<td>23.125</td>
<td>145.30</td>
<td>42</td>
<td>185.00</td>
<td>1,162.4</td>
<td>F7</td>
<td>78</td>
<td>1,480.0</td>
<td>9,299.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G0</td>
<td>7</td>
<td>24.500</td>
<td>153.93</td>
<td>G3</td>
<td>43</td>
<td>196.00</td>
<td>1,231.5</td>
<td>G6</td>
<td>79</td>
<td>1,568.0</td>
<td>9,851.9</td>
</tr>
<tr>
<td>8</td>
<td>25.957</td>
<td>163.09</td>
<td>44</td>
<td>207.65</td>
<td>1,304.7</td>
<td>G7</td>
<td>80</td>
<td>1,611.2</td>
<td>10,428</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A0</td>
<td>9</td>
<td>27.500</td>
<td>172.59</td>
<td>A3</td>
<td>45</td>
<td>220.00</td>
<td>1,382.3</td>
<td>A6</td>
<td>81</td>
<td>1,760.0</td>
<td>11,058</td>
</tr>
<tr>
<td>10</td>
<td>29.135</td>
<td>183.06</td>
<td>46</td>
<td>233.08</td>
<td>1,464.5</td>
<td>A7</td>
<td>82</td>
<td>1,864.7</td>
<td>11,716</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B0</td>
<td>11</td>
<td>30.868</td>
<td>193.95</td>
<td>B3</td>
<td>47</td>
<td>246.94</td>
<td>1,551.6</td>
<td>B6</td>
<td>83</td>
<td>1,975.5</td>
<td>12,413</td>
</tr>
<tr>
<td>C1</td>
<td>12</td>
<td>32.703</td>
<td>205.48</td>
<td>C4</td>
<td>48</td>
<td>261.63</td>
<td>1,643.8</td>
<td>C7</td>
<td>84</td>
<td>2,093.0</td>
<td>13,151</td>
</tr>
<tr>
<td>13</td>
<td>34.648</td>
<td>217.70</td>
<td>49</td>
<td>277.18</td>
<td>1,741.6</td>
<td>C7</td>
<td>85</td>
<td>2,217.5</td>
<td>13,933</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D1</td>
<td>14</td>
<td>36.708</td>
<td>230.64</td>
<td>D4</td>
<td>50</td>
<td>293.66</td>
<td>1,845.2</td>
<td>D7</td>
<td>86</td>
<td>2,349.3</td>
<td>14,761</td>
</tr>
<tr>
<td>15</td>
<td>38.891</td>
<td>244.36</td>
<td>51</td>
<td>311.13</td>
<td>1,954.9</td>
<td>D7</td>
<td>87</td>
<td>2,489.0</td>
<td>15,569</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E1</td>
<td>16</td>
<td>41.203</td>
<td>258.89</td>
<td>E4</td>
<td>52</td>
<td>329.63</td>
<td>2,071.1</td>
<td>E7</td>
<td>88</td>
<td>2,637.0</td>
<td>16,569</td>
</tr>
<tr>
<td>F1</td>
<td>17</td>
<td>43.654</td>
<td>274.28</td>
<td>F4</td>
<td>53</td>
<td>349.23</td>
<td>2,194.3</td>
<td>F7</td>
<td>89</td>
<td>2,793.8</td>
<td>17,554</td>
</tr>
<tr>
<td>18</td>
<td>46.249</td>
<td>290.59</td>
<td>54</td>
<td>369.99</td>
<td>2,324.7</td>
<td>F7</td>
<td>90</td>
<td>2,960.0</td>
<td>18,598</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G1</td>
<td>19</td>
<td>48.999</td>
<td>307.87</td>
<td>G4</td>
<td>55</td>
<td>392.00</td>
<td>2,463.0</td>
<td>G6</td>
<td>91</td>
<td>3,136.0</td>
<td>19,704</td>
</tr>
<tr>
<td>20</td>
<td>51.913</td>
<td>326.18</td>
<td>56</td>
<td>415.30</td>
<td>2,609.4</td>
<td>G6</td>
<td>92</td>
<td>3,322.4</td>
<td>20,875</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A1</td>
<td>21</td>
<td>55.000</td>
<td>345.58</td>
<td>A4</td>
<td>57</td>
<td>440.00</td>
<td>2,764.6</td>
<td>A7</td>
<td>93</td>
<td>3,520.0</td>
<td>22,117</td>
</tr>
<tr>
<td>22</td>
<td>58.270</td>
<td>366.12</td>
<td>58</td>
<td>466.16</td>
<td>2,929.0</td>
<td>A7</td>
<td>94</td>
<td>3,729.3</td>
<td>23,432</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B1</td>
<td>23</td>
<td>61.735</td>
<td>387.90</td>
<td>B4</td>
<td>59</td>
<td>493.88</td>
<td>3,103.2</td>
<td>B7</td>
<td>95</td>
<td>3,951.1</td>
<td>24,825</td>
</tr>
<tr>
<td>C2</td>
<td>24</td>
<td>65.406</td>
<td>410.96</td>
<td>C5</td>
<td>60</td>
<td>523.25</td>
<td>3,287.7</td>
<td>C8</td>
<td>96</td>
<td>4,186.0</td>
<td>26,301</td>
</tr>
<tr>
<td>25</td>
<td>69.296</td>
<td>435.40</td>
<td>61</td>
<td>554.37</td>
<td>3,485.3</td>
<td>C8</td>
<td>97</td>
<td>4,434.9</td>
<td>27,885</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D2</td>
<td>26</td>
<td>73.416</td>
<td>461.29</td>
<td>D6</td>
<td>62</td>
<td>587.33</td>
<td>3,690.3</td>
<td>D8</td>
<td>98</td>
<td>4,698.6</td>
<td>29,522</td>
</tr>
<tr>
<td>27</td>
<td>77.782</td>
<td>488.72</td>
<td>63</td>
<td>622.25</td>
<td>3,909.7</td>
<td>D8</td>
<td>99</td>
<td>4,976.0</td>
<td>31,278</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E2</td>
<td>28</td>
<td>82.407</td>
<td>517.78</td>
<td>E5</td>
<td>64</td>
<td>659.26</td>
<td>4,142.2</td>
<td>E8</td>
<td>100</td>
<td>5,274.0</td>
<td>33,138</td>
</tr>
<tr>
<td>F2</td>
<td>29</td>
<td>87.307</td>
<td>548.57</td>
<td>F5</td>
<td>65</td>
<td>698.46</td>
<td>4,388.5</td>
<td>F8</td>
<td>101</td>
<td>5,587.7</td>
<td>35,108</td>
</tr>
<tr>
<td>30</td>
<td>92.499</td>
<td>581.19</td>
<td>66</td>
<td>739.99</td>
<td>4,649.5</td>
<td>G8</td>
<td>102</td>
<td>5,919.9</td>
<td>37,196</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G2</td>
<td>31</td>
<td>97.999</td>
<td>615.74</td>
<td>G5</td>
<td>67</td>
<td>783.99</td>
<td>4,926.0</td>
<td>G8</td>
<td>103</td>
<td>6,271.9</td>
<td>39,408</td>
</tr>
<tr>
<td>32</td>
<td>103.83</td>
<td>652.36</td>
<td>68</td>
<td>830.61</td>
<td>5,218.9</td>
<td>G8</td>
<td>104</td>
<td>6,644.9</td>
<td>41,751</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A2</td>
<td>33</td>
<td>110.00</td>
<td>691.15</td>
<td>A5</td>
<td>69</td>
<td>880.00</td>
<td>5,529.2</td>
<td>A8</td>
<td>105</td>
<td>7,040.0</td>
<td>44,234</td>
</tr>
<tr>
<td>34</td>
<td>116.54</td>
<td>732.52</td>
<td>70</td>
<td>932.33</td>
<td>5,858.0</td>
<td>A8</td>
<td>106</td>
<td>7,458.6</td>
<td>46,864</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B2</td>
<td>35</td>
<td>123.47</td>
<td>775.79</td>
<td>B6</td>
<td>71</td>
<td>987.77</td>
<td>6,206.3</td>
<td>B8</td>
<td>107</td>
<td>7,902.1</td>
<td>49,651</td>
</tr>
</tbody>
</table>

Numerous subscript notations have been employed to distinguish the notes of one octave from those of another. The particular scheme used here assigns to C0 a frequency, which corresponds roughly to the lowest audible pitch. S is the number of semitones counted from this C0 (1 semitone = 100 Cents).
9.11 Musical Intervals in Cents

Table 16: Intervals in Cents corresponding to certain frequency ratios

<table>
<thead>
<tr>
<th>Name of interval</th>
<th>Frequency ratio (x)</th>
<th>Cents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unison</td>
<td>1:1</td>
<td>0</td>
</tr>
<tr>
<td>Minor second or semitone</td>
<td>1.059463:1</td>
<td>100</td>
</tr>
<tr>
<td>Semitone</td>
<td>16:15</td>
<td>111.731</td>
</tr>
<tr>
<td>Minor tone or lesser whole tone</td>
<td>10:9</td>
<td>182.404</td>
</tr>
<tr>
<td>Major second or whole tone</td>
<td>1.122462:1</td>
<td>200</td>
</tr>
<tr>
<td>Major tone or greater whole tone</td>
<td>9:8</td>
<td>203.910</td>
</tr>
<tr>
<td>Minor third</td>
<td>1.189207:1</td>
<td>300</td>
</tr>
<tr>
<td>Minor third</td>
<td>6:5</td>
<td>315.641</td>
</tr>
<tr>
<td>Major third</td>
<td>5:4</td>
<td>386.314</td>
</tr>
<tr>
<td>Major third</td>
<td>1.259921:1</td>
<td>400</td>
</tr>
<tr>
<td>Perfect fourth</td>
<td>4:3</td>
<td>498.045</td>
</tr>
<tr>
<td>Perfect fourth</td>
<td>1.334840:1</td>
<td>500</td>
</tr>
<tr>
<td>Augmented fourth</td>
<td>45:32</td>
<td>590.224</td>
</tr>
<tr>
<td>Augmented fourth</td>
<td>1.414214:1</td>
<td>600</td>
</tr>
<tr>
<td>Diminished fifth</td>
<td>1.414214:1</td>
<td>600</td>
</tr>
<tr>
<td>Diminished fifth</td>
<td>64:45</td>
<td>609.777</td>
</tr>
<tr>
<td>Perfect fifth</td>
<td>1.498307:1</td>
<td>700</td>
</tr>
<tr>
<td>Perfect fifth</td>
<td>3:2</td>
<td>701.955</td>
</tr>
<tr>
<td>Minor sixth</td>
<td>1.587401:1</td>
<td>800</td>
</tr>
<tr>
<td>Minor sixth</td>
<td>8:5</td>
<td>813.687</td>
</tr>
<tr>
<td>Major sixth</td>
<td>5:3</td>
<td>884.359</td>
</tr>
<tr>
<td>Major sixth</td>
<td>1.681793:1</td>
<td>900</td>
</tr>
<tr>
<td>Harmonic minor seventh</td>
<td>7:4</td>
<td>968.826</td>
</tr>
<tr>
<td>Grave minor seventh</td>
<td>16:9</td>
<td>996.091</td>
</tr>
<tr>
<td>Minor seventh</td>
<td>1.781797:1</td>
<td>1,000</td>
</tr>
<tr>
<td>Minor seventh</td>
<td>9:5</td>
<td>1,017.597</td>
</tr>
<tr>
<td>Major seventh</td>
<td>15:8</td>
<td>1,088.269</td>
</tr>
<tr>
<td>Major seventh</td>
<td>1.887749:1</td>
<td>1,100</td>
</tr>
<tr>
<td>Octave</td>
<td>2:1</td>
<td>1,200.000</td>
</tr>
</tbody>
</table>

Note: Cents = \[1200 \log_{10}(f_1/f_2)] / \log_{10}(2)\) and \(x = (f_1 / f_2) = 2^{(\text{Cents}/1200)}\)
## 9.12 Hotkeys

### 9.12.1 Waveform Viewer1 Hotkeys

Table 17. Waveform Viewer1 Hotkeys

<table>
<thead>
<tr>
<th>Hotkey</th>
<th>File-Overview Window</th>
<th>Segment (Line) Window(s)</th>
<th>Zoom Window(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Ins]</td>
<td>create soundsegment bracketed by cursors</td>
<td>(not applicable)</td>
<td></td>
</tr>
<tr>
<td>[Space]</td>
<td>play signal window before – after active cursor position</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Esc]</td>
<td>close active dialogs – stop play immediately</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[←],[→]</td>
<td>move active cursor left/right</td>
<td>move active cursor left/right and wrap around line(s)</td>
<td>move active cursor left/right</td>
</tr>
<tr>
<td>Shift-[←],[→]</td>
<td>move active cursor left/right (in larger steps)</td>
<td>move active cursor left/right (in larger steps)</td>
<td>move active cursor left/right (in larger steps)</td>
</tr>
<tr>
<td>[1],[.]</td>
<td>(not applicable)</td>
<td>move active cursor to previous/next line</td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[Pos1],[End]</td>
<td>move active cursor to begin/end of file</td>
<td>move active cursor to begin/end of line</td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[Page Up]</td>
<td>(not applicable)</td>
<td>cursor to first line</td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[F2] (cyclic)</td>
<td>activate first/next cursor</td>
<td></td>
<td>2 cursors only1</td>
</tr>
<tr>
<td>[F3] (toggle)</td>
<td>mirror inactive cursor around active cursor position / back to its original position</td>
<td></td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[Ctrl-F3] (toggle)</td>
<td>mirror active cursor around inactive cursor position / back to its original position</td>
<td></td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[F4]</td>
<td>increase play window length (double length)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Ctrl-F4]</td>
<td>decrease play window length (half length)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[L] (toggle)</td>
<td>lock/unlock cursor distance</td>
<td>(not applicable)</td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[P]</td>
<td>play signal bracketed by cursors</td>
<td></td>
<td>2 cursors only1</td>
</tr>
<tr>
<td>[Q]</td>
<td>play soundfile</td>
<td>play soundsegment</td>
<td>play window</td>
</tr>
<tr>
<td>[C] (cyclic)</td>
<td>change cursor mode</td>
<td>(not applicable)</td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[S]</td>
<td>(not applicable)</td>
<td>snap to marked segment next to active cursor2</td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[X],[Y]</td>
<td>(not applicable)</td>
<td>go to next/previous marked segment2</td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[Ctrl-Y]</td>
<td>(not applicable)</td>
<td>go to 1st marked segment2</td>
<td>(not applicable)</td>
</tr>
<tr>
<td>[Ctrl-S]</td>
<td>(not applicable)</td>
<td>save (= set beginning and end of selected marked segment to current cursor positions)2</td>
<td>(not applicable)</td>
</tr>
</tbody>
</table>

1 Only if 2 zoom cursors are displayed (zoom mode = "Between Cursors")
2 Available only if segment markers are displayed
9.13 Random Number Generators

9.13.1 MathCad

Program to return a matrix of Normally Distributed Random Numbers

\[
M(n, \mu, \sigma) = \begin{cases} 
  b \leftarrow 0 \\
  V \leftarrow \text{norm}(m, \mu, \sigma) \\
  \text{return } V \text{ if } n = 1 \\
  \text{while } b < n \rightarrow 1 \\
  V \leftarrow \text{augment}(\text{norm}(m, \mu, \sigma), V) \\
  b \leftarrow b + 1 \\
  V 
\end{cases}
\]

\[
M(10, 0) = \begin{pmatrix}
  0 & 1.084 & 0.983 & 1.128 & 0.922 & 1.019 & 1.015 & 1.027 & 0.856 & 1.042 \\
  1 & 0.909 & 1.159 & 0.975 & 0.799 & 1.032 & 0.947 & 1.017 & 1.001 & 0.989 & 0.944 \\
  2 & 0.967 & 1.001 & 1.046 & 1.070 & 1.068 & 0.972 & 0.866 & 0.946 & 0.938 & 1.066 \\
  3 & 0.888 & 0.935 & 1.13 & 0.974 & 1.031 & 1.365 & 0.967 & 1.081 & 0.918 & 1.066 \\
  4 & 1.031 & 1.026 & 0.988 & 0.942 & 1.025 & 0.812 & 1.026 & 1.048 & 0.825 & 0.979 \\
  5 & 1.047 & 1.102 & 0.992 & 1.058 & 1.103 & 1.047 & 1.042 & 0.744 & 0.952 & 0.907 \\
  6 & 0.933 & 1.029 & 1.078 & 0.831 & 0.965 & 1.065 & 0.942 & 1.063 & 1.072 & 0.756 \\
  7 & 0.936 & 0.936 & 0.862 & 0.991 & 0.903 & 0.926 & 1.018 & 0.924 & 0.886 & 1.240 \\
  8 & 1.021 & 0.884 & 0.983 & 1.017 & 1.141 & 0.939 & 1.134 & 1.164 & 0.974 & 1.052 \\
  9 & 0.918 & 1.033 & 1.038 & 0.845 & 0.907 & 0.997 & 1.016 & 1.037 & 1.046 & 0.953 
\end{pmatrix}
\]
9.14 Glossary

**Acoustic energy**: Variation in air pressure produced by the vibration of an object.

**AES/EBU interface**: Digital audio transmission interface standard, developed by the Standards Committee of the Audio Engineering Society and adopted by the European Broadcasting Union.

**Amplitude of a sound wave**: Maximum pressure variation, a force per unit area measured in Newtons per square meter over time.

**Analog to Digital conversion**: The analogue input signal is converted into a stream of numeric values (A/D conversion).

**Analytic listening**: Listening for the pure tone components of a complex tone, as opposed to holistic (synthetic) listening.

**Audigram**: Graph of threshold intensity for hearing pure tones as a function of frequency.

**Auditory cortex**: Region of the cortex devoted to the analysis of sound information.

**Auditory nerve**: Bundle of nerve fibres which carries information from the cochlea to the higher stages of the auditory system.

**Auditory stream**: Sequence of sounds grouped together because they are attributed to the same source (sonic event). Note that Warren uses the term “parallel auditory continua”.

**Basilar membrane**: A membrane inside the cochlea which vibrates in response to sound. It is here that sound energy is converted into neural impulses.

**Chord**: Three or more notes sounded simultaneously.

**Chroma**: Musical note name, without specification of the octave register.

**Chromatic scale**: A scale of twelve equal steps per octave, each step being a semitone.

**Cochlea**: A coiled, fluid-filled chamber in the inner ear, containing the basilar membrane, where mechanical (sound) energy is converted into neural energy. Resolution of a complex sound into its components occurs in the cochlea.

**Complex tone**: A tone composed of a number of sinusoids at different frequencies and phases (not necessarily harmonic).

**Component**: One of the sinusoid that is part of a complex sound.

**Cycle**: That part of a periodic function that occurs in one period.

**dB (Decibel)**: A dimensionless unit for the expression of levels of power and amplitude measurements such as the sound pressure level (SPL):

\[ L = 20 \cdot \log_{10} \left( \frac{p}{p_0} \right) dB \]

L = level of the sound pressure in dB(SPL)
Appendix

\[ p = \text{sound pressure measured in } \text{N/m}^2 \]
\[ p_0 = 2 \cdot 10^{-5} \text{ N/m}^2 = \text{reference sound pressure}. \]

**Difference threshold:** Minimum amount by which stimulus intensity must be changed in order to produce a just noticeable change in sensation.

**Equal loudness contours:** Curves plotted as a function of frequency, showing the sound pressure level required to produce a given loudness level.

**Envelope:** The envelope of a function is a smooth curve passing through the peaks of the function. e.g. spectral envelope.

**FFT:** Fast Fourier Transform. A specific efficient spectral analysis program.

**Formant:** A resonance in the vocal tract, which causes a peak in the spectral envelope of a speech sound.

**Frequency:** For a sine wave, the frequency is the number of repetitions per unit of time. 1 cycle per second = 1 hertz. Usually referred to in kHz, or number of repetitions per ms.

**Fundamental:** Lowest component of a harmonic complex tone.

**Fundamental frequency:** The fundamental frequency of a periodic sound is the frequency of repetition of the waveform.

**Habituation:** The process by which an organism ceases to respond to some recurring or familiar stimulus.

**Harmonic:** A component of a complex tone, whose frequency is an integral multiple of the fundamental frequency of the complex. The third harmonic is at the frequency 3f, where f is the fundamental frequency.

**Harmonic complex tone:** Complex tone whose partials are all harmonics i.e. all partials have frequencies that are integral multiples of the fundamental frequency. Harmonic complex tones are periodic.

**Hear out:** Hear, by careful analytic listening, the components of a complex tone.

**Hertz:** Unit of frequency. 1 Hz = 1 cycle per second.

**Holistic listening:** Normal mode of perceiving the whole without being aware of the components of a complex tone. Opposite of analytical. Also called synthetic.

**Intensity:** Sound power transmitted through a given area. Intensity is proportional to the square of the amplitude and, expressed as power per unit area, it is measured in watts per square meter. The threshold of audibility is 10 power -12 W/sq m.

**Logarithmic scale:** A scale in which the logarithm of the raw value is used instead of the raw value. The effect is that equal steps in the raw value are replaced by equal ratios e.g. dB scale.

**Loudness:** Attribute of auditory sensation corresponding to intensity. Sounds can be ordered on a loudness scale from quiet to loud.
Appendix

**Mel scale:** A proportional scale, in which equal intervals (measured in mel) correspond to equal perceived interval sizes. The mel scale is roughly proportional to the logarithm of frequency but becomes linear at low frequencies.

**Noise:** Usually refers to unwanted sound. Noise is not periodic.

**White noise:** is a sound with constant power per unit bandwidth over the audible frequency range.

**Octave:** The interval between two tones when their frequencies are in the ratio 2:1. Musical notes an octave apart have the same letter name.

**Octave-related complex tone:** Complex tone whose components are separated by octaves.

**Partial:** One of the sinusoidal components that is part of a complex sound.

**Perfect fifth:** Interval between the first and fifth degrees of a major scale, or interval between two pure tones whose frequencies are in the ratio 3:2, or 7 semitones.

**Perfect fourth:** Interval between the first and fourth degrees of a major scale, or interval between two pure tones whose frequencies are in the ratio 4:3, or 5 semitones.

**Perfect pitch:** See absolute pitch.

**Period:** The smallest time interval over which a function repeats itself.

**Periodic sound:** A periodic sound has a waveform which repeats regularly over time.

**Phase:** Relation in time between two pure tones of the same frequency. In **phase:** both waveforms peak together.

**Pitch:** The attribute of auditory sensation by which sounds can be ordered on a musical scale i.e. by which sounds can be judged relatively high or low.

**Pitch ambiguity:** A sound has pitch ambiguity if holistic perception may yield one of several different pitches depending on attention, context,..

**Pitch shifts:** Change in pitch of a tone due to intensity or masking.

**Proximity:** Gestalt principle of organisation referring to the perceptual tendency to group together objects that are near to one another. In auditory perception, “near” means close in the frequency-time domain.

**Psychophysics:** Branch of Perception that is concerned with establishing quantitative relations between physical stimulation and perceptual events.

**Pure tone:** A tone whose sound wave is sinusoidal.

**Relative pitch:** Ability to identify a musical interval. An ability which can be learned.

**Residue pitch:** Also known as low pitch, virtual pitch, periodicity pitch. Pitch of a complex tone, under normal listening. The name “residue” refers to the phenomenon of the missing fundamental -the pitch perceived comes from the “residue”.
Appendix

**Salience**: Perceptual prominence, or likelihood of being noticed.

**Semitone**: The notes corresponding to adjacent keys on the piano. The interval that results when the octave is divided into 12 equal intervals.

**Similarity**: Gestalt principle of organisation referring to the perceptual tendency to group together objects that are similar in texture, size, shape, pitch, loudness, timbre,...

**Sine wave**: A waveform whose variation over time is the sine function. The most efficient form of oscillating motion.

**Spectral analysis**: Mathematical analysis of a waveform into sinusoidal components, as by Fourier analysis.

**Spectral dominance**: Effect by which the pure tone components in the range 200 Hz to 2000 Hz have the greatest influence on the perception of a complex sound.

**The spectral dominance**: function (averaged over many subjects) for pure tones has a broad peak around 700 Hz.

**Spectrum**: A spectrum graph shows the power (or amplitude) in each of the component frequencies.

**Timbre**: Relates to the quality of a sound. Timbre depends on the frequency and amplitude of partials, and on their evolution over time.

**Tone**: A sound wave that evokes a sensation of pitch.

**Virtual pitch**: The pitch of a complex tone with synthetic (holistic) listening. See residue pitch.
9.15 Some Examples of Speech Analysis

Figure 66: Speech Analysis Setup.
Appendix

Figure 67: Example: Speech Analysis.
Appendix

Stx Spectral Analysis: Wavelet - Spectrogram, Setup for Speech
Appendix

Spectrogram Speech: Waveform, f0 Formant-Candidates
10. Table of Figures

Figure 1: The multi-layer segmentation system for soundfiles. Sound segment addresses, segment identifiers, optional links and content descriptions (annotations) are stored in a separate reference data file called a SOUNDSEGMENT DIRECTORY file, normally located in the same folder as the current soundfile.

Figure 2: Soundsegment directory. From left to right: SegmentName StartTime Duration Annotation (optional).

Figure 3: The STx main window. At first startup of STx it appears in the upper left corner of your screen. Below it, the empty SOUNDFILE LIST window appears.

Figure 4: The empty S_TOOLS SOUNDFILE LIST (SFList) window at first startup of STx

Figure 5: The tools/applications available from the STx main window. By default, and at first startup, the S_Tools application is started, which opens the SOUNDFILE LIST window.

Figure 6: The STx soundfile work area as accessed through the SOUNDFILE LIST, and the 5 submenus which control it

Figure 7: The Structure of the STx Soundfile Work Area: The soundfile list window and segment list windows (soundsegment directories) of the open soundfiles.

Figure 8: Record window and Record Setup dialog. The default settings shown here create a stereo soundfile, sampling frequency 44100 Hz, binary wordlength 16 bit. The default location of the temporary soundfile is drive D:A_Trans.

Figure 9: Mixer settings for analog or digital input (SPDIF-IN) of a Creative Soundblaster Live sound card under Windows.

Figure 10: Record Level Control: STx shows the current peak level in real time as well as the history of the current recording session (two channel stereo recording).
Table of Figures

Figure 11: Record Level Control and Cue-Point window. By means of [Shift+left mouse button] on the waveform display a tag is created which can be used as a cue point for defining segments during recording of a soundfile. 18

Figure 12: Play: track#2 of soundfile Neaple.wav is currently being reproduced. The progress bar indicates the current position in the sound segment. 19

Figure 13: Find soundfiles on attached disk drives and Add them to the work area of the current STx session. The pop-up window is opened by clicking the right mouse button inside the list pane. 22

Figure 14: Naming and tagging of soundsegments in a soundfile. Metadata are stored in the associated soundsegment directory file. 26

Figure 15: Soundsegment creation and manipulation functions are accessed via the Segments menu of the soundsegment list window (which is displayed whenever you open a soundfile). 27

Figure 16: Modify. SegmentName, addresses and annotation data can be modified in STx online as well as offline by editing the soundsegment directory files. 27

Figure 17: Example of a segment Sort: Here the sort criteria are Name – Begin – Text invers (=descending). SegmentNames, addresses and annotations can be used for sorting segment lists in soundfile directory files. Up to 3 sort criteria can be combined in a single task. 29

Figure 18: Soundsegment Database: provides access to soundsegments and referenced metadata stored in selected soundfiles and associated soundsegment directory files. 30

Figure 19: Four steps for analysing and displaying a soundsegment: [1] select soundfile from soundfile list window (here: Stockholm.wav), [2] select soundsegment from the soundsegment list, [3] select Analyse from the Signal menu [4] select a specific analysis type and start. 32

Figure 20: Analyse – Waveform 3 Lines: from top to bottom: [1] zoom windows for left/right cursor of the 3-line display (window length +/- 40ms each), [2] 3 waveform traces showing the total length of the segment take#04 (191s), [3] waveform envelope of the complete soundfile Neapel.wav (duration 3451s). Arrows point from the location of the segment take#04 (selected in the previous dialog box) to its beginning and end in the 3-line display. 35

Figure 21: Move the Windows system cursor close to the STx waveform cursor. Whenever the current x/y values (= current time and amplitude reading) are displayed above (top graphic) or below (bottom graphic) the Windows cursor position, the STx cursor (here purple) can be moved by clicking the left mouse button and dragging, or with cursor hotkeys (see Table 3) when the STx cursor is activated (by a left mouse click or F2/Ctrl-F2). 36
Table of Figures

Figure 22: What happens when you press F3: before (left graphic) and after (right graphic). (The active cursor is purple, the inactive cursor turquoise.) 38

Figure 23: What happens when you press Ctrl-F3: before (left graphic) and after (right graphic). (The active cursor is purple, the inactive cursor turquoise.) 38

Figure 24: Waveform zoom windows: zoom windows of left/right cursor (window length +/-40ms each). The cursors of both zoom windows, which by default are located at time zero in the middle of the zoom window, can be moved independently under auditory control in order to identify the exact cue-in and cue-out point of a segment. After adjustment, hotkeys [1] and [2] update the cursors in the waveform line display. 39

Figure 25: [INS]: Segment Naming, Tagging and Annotation window. Default soundfile template. In order to create a new template see 9.1 Defining Soundsegments. 40

Figure 26: Cascading multiple Waveform Viewer windows. In order to cascade multiple Viewer windows, select Cascade from the Process menu in the soundfile list window. 41

Figure 27: Default spectrogram viewer display: Graph 1 – waveform zoom window Graph 2 – amplitude-spectrum display Graph 3 -main spectrogram window Graph 4 – segment waveform display Further graphs can be added, such as the parameter plot in Graph 5. 44

Figure 28: The Analyse startup sequence: [1] select a soundfile from soundfile list window, [2] select a soundsegment from soundsegment list, [3] select Analyse from the Signal menu and [4] choose the appropriate analysis method. Note: if the method configuration and the parameter setup have been selected appropriately, steps [3 to 4] can be replaced by just activating View in step [3]. The program will immediately start the analysis using the current default parameter. 45

Figure 29: creation of a narrowband spectrogram, including analysis parameter setup control. The parameter setup dialog is obtained by double clicking on the selected method; each method has its own independent parameter setup. Currently the default spectrogram bandwidth is set according to FFT-frequency bins (and window length) to 22 Hz, %-overlap = 75 (frame shift) of FFT window. Frequency range = 0 to 10000 Hz, dynamic range = 50 dB up from -70 dB FS. 46

Figure 30: Overlap-add and analysis frame (time) shift. Shifting the (time) window by half of its length enables loss-free reconstruction of the original waveform. Time shifts of > half of the window length will cause loss of information (and so should be used with caution). 47

Figure 31: Narrowband spectrogram of a synthesized Sonata (J.S. Bach): bandwidth according to FFT-frequency bins (and window length) is set to 10 Hz, %-overlap = 50% (frame shift) of FFT window. Frequency range = 0 to 4000 Hz, dynamic range = 50 dB up from -70 dB FS. The left sectioner window (top left) shows the waveform
Table of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>Widedband spectrogram of the utterance ‹enthusiasm› [EnTj'u:ziaz7m], male speaker: bandwidth according to FFT-frequency bins (and window length) is set to 80 Hz, %-overlap = 90 (frame shift) of FFT window. Frequency range = 0 to 8000 Hz, dynamic range = 50 dB up from -70 dB FS. The appearance of vertical bars with spectral dominance in the formant frequency regions represents the periodicity of the voice fundamental frequency as well as the resonance behaviour of the vocal tract during voiced speech sounds. The left sectioner window shows the waveform centred at the active cursor position on the spectrogram, the right sectioner window shows the corresponding FFT magnitude as well as the LPC-smoothed spectrum. The parameter window (middle left) shows the time and frequency cursor values displayed in the sectioner.</td>
</tr>
<tr>
<td>33</td>
<td>Functional diagram of cepstral analysis.</td>
</tr>
<tr>
<td>34</td>
<td>Cepstrally smoothed spectrum (order 32) of a log magnitude spectrum, male voice. Number of low quefrency cepstrum coefficients 32.</td>
</tr>
<tr>
<td>35</td>
<td>Log amplitude spectrum of a voiced speech segment, male voice. LPC smoothed spectrum superimposed. Number of LP coefficients (p) equals the sampling frequency of the signal in kHz plus 4 (&quot;0&quot;). The peaks of the LPC smoothed spectrum are used as formant frequency candidates.</td>
</tr>
<tr>
<td>36</td>
<td>Standard Broadband Spectrogram, LPC-formant candidates, fundamental frequency contour, waveform envelope and sectioner windows. Default setting of Viewer2, procedure Speech Analysis.</td>
</tr>
<tr>
<td>37</td>
<td>Viewer2-Pseudo Wigner Distribution spectrograms, comparable in frequency resolution to broadband FFT-analysis. Default Setup see below.</td>
</tr>
<tr>
<td>39</td>
<td>Frequency Band Signal Energy: the middle trace shows the 10-band signal energy as a function of time. The frequency bands can be defined in the Setup dialog window (shown in the upper right corner).</td>
</tr>
<tr>
<td>40</td>
<td>Linear averaging of the quasi steady state part of the phoneme /ea/. The LPC smoothed spectrum is overlaid on the FFT amplitude spectrum.</td>
</tr>
<tr>
<td>41</td>
<td>Linear averaged amplitude spectrum of the quasi steady state part of the phoneme /ea/ posted in the Spectrum Editor section of STx DataSet. The amplitude spectrum has been inverted.</td>
</tr>
</tbody>
</table>
Table of Figures

Figure 42: The Process Segment(s) dialogue box provides the main entry for setup and start of STx Signal Processing functions currently available. Note: these functions are intentionally changing the content of soundfile(s). 60

Figure 43: Default settings of the AGC function: The user is invited to start with these default settings. As no generally applicable rules can be given, optimisation of the AGC processing has to remain empirical. 61

Figure 44: functional diagram of the STx denoising module using spectral averaging and spectral subtraction. 63

Figure 45: The Select a Sequence window when the first segment has been selected to create one’s first sequence. 65

Figure 46: The first segment (here: My1stSegment, from the soundfile F:\work\RB2.wav) of the first sequence created. 66

Figure 47: The Select a Sequence window when a first sequence already exists. Click on Select to add the soundsegment to sequence "#1" or New to create a new sequence. 66

Figure 48: A sequence containing multiple soundsegments from two different soundfiles. 66

Figure 49: This sequence is composed of synthesized signals as well as of segments taken from a soundfile. 67

Figure 50: Spectrogram of a sequence containing multiple synthesized sounds and segments from a soundfile. 68

Figure 51: Opening of the DataSet from the Tools pop down menu. 69

Figure 52: Speech Spectrogram with formant frequency candidates overlaid and fundamental frequency contour containing downward octave jumps due to high pitched male voice; voice setting close to singing, music instrument tone background (guitar). 70

Figure 53: List of DataSet entries of the current STx session. 71

Figure 54: View starts the display and Edit section of the DataSet Editor after selection of the parameter sets to be edited (toggle ON/OFF). 71

Figure 55: Corrections of wrong values of fundamental frequency raw data by means of the interactive DataSet editor. Values in double line frame boxes are corrected by multiplying with the factor 2. Single line frame box values are set to zero. 72

Figure 56: The corrected and cleaned fundamental frequency contour is ready for statistical post processing. 72
Table of Figures

Figure 57: Basic statistics (available in the STx DataSet) of the corrected fundamental frequency contour of above. 73

Figure 58: Input window for a new entry in the segment directory: SegmentName (Name) SegmentStartAddress (Begin/Task) SegmentDuration (Length) and optional annotations (here: Typ, Text and Tag). 75

Figure 59: open the soundsegment directory file template management. 77

Figure 60: soundsegment directory files can be configured to meet the requirements of virtually all database systems. The number of fields for annotation can be extended to any size necessary. 78

Figure 61: Waveform Display – Viewer 1. Checking timebar only speeds up the waveform display for signals of long duration. 80

Figure 62: Analysis – Viewer 2: %-overlap reduced to 60% speeds up the spectrogram display for narrowband (10 Hz) frequency analysis. 81

Figure 63: Peak-to-peak, RMS and AVG values of sinusoidal waveforms. 83

Figure 64: Peak-to-Peak, RMS and AVG values of simple waveforms. 83

Figure 65: Peak-to-Peak, RMS and AVG values of special waveforms. 83

Figure 66: Speech Analysis Setup. 93

Figure 67: Example: Speech Analysis. 94